

EVANSVILLE REGIONAL TRAVEL MODEL DEVELOPMENT

**Technical Memorandum:
*Automated Air Quality Conformity Analysis***

Prepared for the

EVANSVILLE URBAN TRANSPORTATION STUDY

Room 316, Civic Center Complex
Evansville, Indiana 47708
(812) 436-7833

September 2003

Prepared by

BERNARDIN-LOCHMUELLER & ASSOCIATES, INC.
6200 Vogel Road
Evansville, IN 47715
(812) 479-6200 (800) 423-7411 (812) 479-6262 FAX

TABLE OF CONTENTS

Documentation

Introduction	2
Estimation of Hourly Speeds and Volumes (POST_ALT)	4
Preprocessing for Mobile 6	5
Automated Mobile 6 Runs	6
Postprocessing of Mobile 6	7
Appendix A: Mobile 6 Control Input File	9
Appendix B: Example Mobile 6 External Data Input Files	13
Appendix C: Mobile 6 Report Output File	16
Appendix D: Associated GIS/DK Macros Code	31

Introduction

The Clean Air Act (CAA) and the Transportation Equity Act for the 21st Century (TEA-21) require evaluation of transportation plans for areas that are designated as “non-attainment” or “maintenance” areas for the National Ambient Air Quality Standards (NAAQS) to ensure consistency with air quality planning efforts. Certain activities require that a conformity determination be made, namely the development of new or amended long-range transportation plans or short-range transportation improvement programs (TIP) for Urbanized Areas.

To assist in required air quality conformity analyses, the Evansville Urban Transportation Study (EUTS) contracted with Bernardin-Lochmueller & Associates (BLA) to develop a micro-computer program to interface with and post-process the output of the new Evansville Regional Travel Model, also developed by BLA for EUTS.

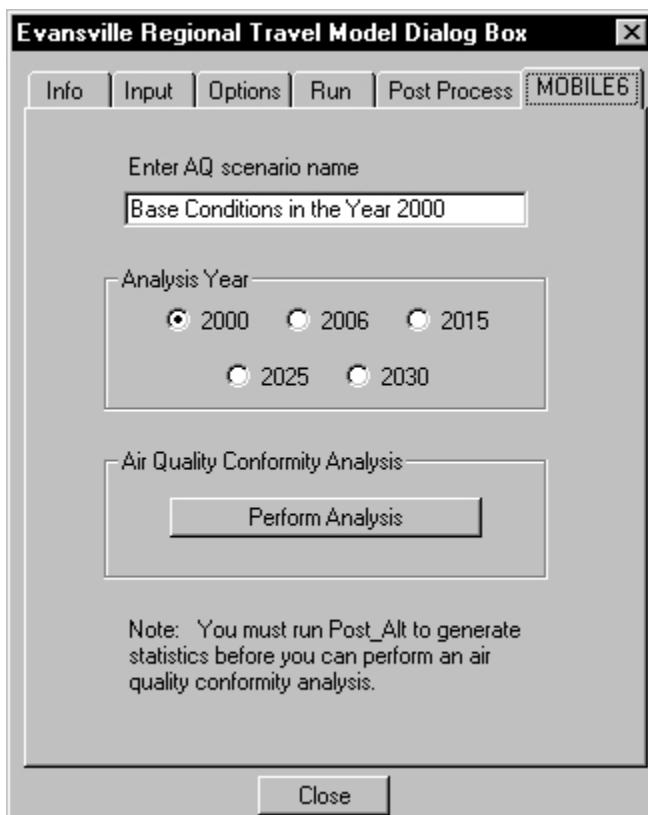


Figure 1: Air Quality Post-Processor User Interface

The air quality post-processor program was written in the GIS/DK programming language and incorporated into the travel model's user interface in the TransCAD software environment. This enables the user to run the air quality program after making a travel model run by the simple push of a button. The user needs only to specify the year of the analysis, and the program incorporates the parameters specific to the analysis area in that year. The user interface is displayed in Figure 1. The current program is designed only to determine conformity for Vanderburgh County, since it is

currently the only non-attainment area in the Evansville region. If due to new standards (e.g., the 8-hr ozone standard, etc.) or for other reasons future conformity analyses require the examination of other counties within the model area, the program could be modified in a fairly simple, straightforward manner to report emissions and conformity for other counties or the model area as a whole.

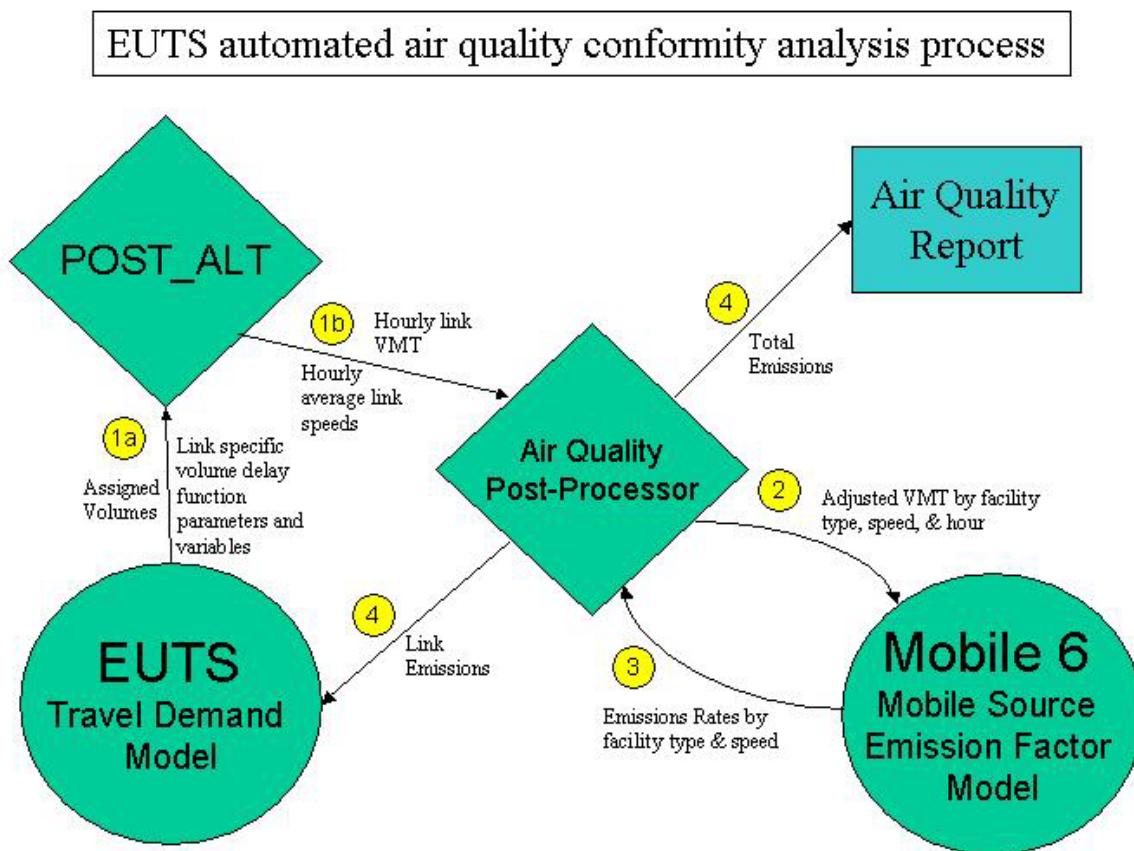


Figure 2: EUTS Automated Air Quality Conformity Analysis Process

The process of air quality conformity analysis automated by the post-processor is displayed in Figure 2. The process can be understood as being made up of four steps. The first step is the estimation of hourly link speeds and volumes. This step is accomplished by the POST_ALT post-processor for the travel model that also produces other traffic statistics. In the second step, the air quality post-processor (proper) adjusts and aggregates the link VMT from POST_ALT by facility type, speed, and hour of the day and passes this data along with the other standard inputs to the Mobile 6 model. The third step would be the series of thirty Mobile 6 runs to produce emissions rates by facility type and speed bin. The fourth and final step then consists of application of the rates from the Mobile 6 runs to the VMT to calculate the emissions and the reporting of these emissions on the network links and in summary in the report file.

Estimation of Hourly Speeds and Volumes

The average congested speeds for links in the model's roadway network used by the air quality post-processor are computed first by a post-processor for the travel model, POST_ALT. POST_ALT, developed by BLA, passes to the air quality post-processor the hourly VMT and average speeds on each link in the model's roadway network. The air quality post-processor then tabulates the VMT on each facility type, in each speed bin, for each hour of the day. Harmonic means are used in disaggregating each link's hourly VMT into the upper and lower bins around its hourly average speed, as described in section 5.3.4 of the Mobile 6 User's Guide.

The hourly average speed for each link is calculated by using the traditional Bureau of Public Roads (BPR) volume delay function. Link specific parameters are used to adjust the link's free-flow speed on the basis of its hourly volume to capacity ratio to account for congestion related delay. The alpha and beta parameters for the BPR equation which are used in both the travel model's assignment procedure as well as the post-processing are coded on the network links. Two sets of parameters were used in the EUTS model. Rural principal arterials and interstates were coded with an alpha of 0.83 and beta of 2.10. All other links were coded with an alpha of 0.84 and a beta of 5.50. These parameters are taken from *Delay-Volume Relations for Travel Forecasting, Based on the 1985 Highway Capacity Manual* by Alan Horowitz, published by the FHWA in 1991 and cited by the Travel Model Improvement Program's (TMIP) 1997 report on *Travel Model Speed Estimation and Post Processing Methods for Air Quality Analysis*. The link free-flow speeds were assigned on the basis of posted speed, functional class, number of lanes, and presence/absence of median; and on some facilities these free-flow speeds may have been subsequently adjusted in the calibration process. The speed table used in assigning speeds was developed on the basis of an extensive speed study conducted at over 60 stations in southwest Indiana for the study of I-69. The capacities used in the estimation of average speeds are absolute (LOS E) capacities from the Highway Capacity Manual. The last input to the volume delay function, the volume, is estimated by apportioning the model's assigned daily volumes using an hourly distribution and peak-period directional factor that are specified by the user. The examples given in this documentation all reflect the use of the a 60/40 directional split in the peak periods and an hourly distribution obtained by averaging the trip distributions reported in the 2000 Evansville Household Travel Survey and the 1995 Indiana Statewide Household Travel Survey.

A formal calibration/validation of speeds was not part of the BLA's services contracted to EUTS, nor was there average speed data available for the area necessary for such a process. Therefore, the validation of the average speeds was limited to reasonableness checks. However, POST_ALT has recently been calibrated for another urban model, using local peak period speed studies, and obtained a 36.1% RMSE with AM period speeds and a 31.5% RMSE with PM period speeds. This calibration was accomplished by dampening the hourly distribution from a household travel survey to account for a more disperse commercial traffic distribution and disproportionate underreporting of off-peak period trips. Averaging the distributions from the Evansville and Indiana statewide

household travel surveys produced a fairly similar distribution due to the less peaked character of the statewide distribution. Another hourly distribution of traffic for the Evansville region was produced by dampening the distribution from the 2000 Evansville Household Travel Survey similarly as was done to achieve the calibration in the other model. It could be appropriate to use either of these distributions (the average or the dampened distribution), but statistical agreement of the calculated speeds with observed speeds is not expected to be as good here because no observed speed data was available for calibration purposes.

Preprocessing for Mobile 6

The preprocessor for the thirty Mobile 6 runs produces a control file to implement the Mobile 6 runs and creates fifty-one external data files. The creation of the control file is very straightforward since most of the assumptions for the Mobile 6 runs can be hard-coded because they depend only on the air quality analysis area and do not vary from analysis to analysis. (These assumptions are recorded in the next section dealing with the Mobile 6 runs.) A representative Mobile 6 input control file created in the preprocessor is reproduced in Appendix A.

The external data files created in the preprocessor are distributions of VMT in the analysis area by facility type, speed bin, and hour of the day. Examples of these external data files produced in the preprocessor can be found in Appendix B. The preprocessor essentially cross tabulates the hourly link VMT's for each combination of facility type, speed bin, and hour of the day and then converts these sums into distributions. The Mobile 6 facility type is coded directly on the model's roadway network links in the field M6FT. Freeways are coded as 1, arterials (including collectors) by 2, locals by 3, and any ramps by 4 in the M6FT field. However, before the VMT can be tabulated, it must be adjusted to ensure relative agreement with HPMS and account for the underrepresentation of local streets in the model.

Year 2000 VMT	Model		Model Adjusted		HPMS	
	VMT	Share	VMT	Share	VMT	Share
Freeways	508,579	12.3%	508,579	11.6%	432,400	9.9%
Arterials	3,365,478	81.5%	3,365,478	76.6%	3,443,000	78.4%
Locals	213,065	5.2%	473,538	10.8%	476,000	10.8%
Ramps	44,224	1.1%	44,224	1.0%	37,600	0.9%
Total	4,131,347	100.0%	4,391,819	100.0%	4,389,000	100.0%

Table 1: HPMS vs. Model VMT Table

For each facility type, as well as for Vanderburgh County as a whole, the daily VMT reported by the travel model for the year 2000 was compared with the daily VMT estimates from the Federal Highway Administration's (FHWA) Highway Performance Monitoring System (HPMS). Because the travel model's roadway network does not contain ramps, the Mobile 6 national default was used to break out the model's total freeway VMT into mainline and ramp VMT. Because the model network also includes only a fraction of the local streets, it was expected that the VMT for local facilities would

have to be factored up to yield an accurate estimate of VMT. The comparison revealed that if the VMT reported by the model for local roads was factored up to agree with the HPMS estimate, the total VMT and shares for each facility type were in good agreement between the model and the HPMS estimates. The base year VMT by facility type for Vanderburgh County are displayed in Table 1. The same factor was applied to the local facilities' VMT for future years as was used in the base year of 2000.

Automated Mobile 6 Runs

The air quality post-processor makes thirty Mobile 6 runs for each travel model run in order to produce a table of emissions rates by facility type and speed bin. An example of an emissions rate table produced by the post-processor is displayed in Table 2. Mobile 6 model documentation (User's Guide section 2.8.8.2.d) makes clear that it is not inappropriate to use Mobile 6 to model emissions rates for roadway links separately. Section 4.5.3 of the TMIP's report entitled Travel Model Speed Estimation and Post Processing Methods for Air Quality Analysis also encourages the estimation of link based emissions and emissions rates.

Facility Type		Average Speed (in miles per hour)													
		2.5	5	10	15	20	25	30	35	40	45	50	55	60	65+
Freeway	VOC	12.96	5.46	3.10	2.43	2.11	1.96	1.85	1.76	1.71	1.66	1.58	1.61	1.63	1.59
	CO	58.29	34.50	21.84	18.17	17.10	16.52	16.16	16.19	16.82	17.49	17.12	19.74	22.25	22.81
	NOX	4.37	3.96	3.20	2.79	2.70	2.64	2.62	2.61	2.65	2.72	2.83	3.00	3.23	3.53
Arterial	VOC	12.96	5.11	3.04	2.44	2.11	2.04	1.95	1.84	1.77	1.72	1.68	1.65	1.62	1.59
	CO	56.29	33.14	22.12	18.78	17.13	17.68	17.67	17.71	18.37	19.14	19.99	20.97	21.82	22.65
	NOX	4.04	3.52	3.00	2.66	2.47	2.40	2.34	2.31	2.34	2.41	2.52	2.67	2.89	3.19
Local	VOC				2.86										
	CO				17.84										
	NOX				2.33										
Ramp	VOC								2.07						
	CO								27.90						
	NOX								2.30						

Table 2: Emissions Rate Table

Each of the thirty Mobile 6 runs assumes the same fuel volatility, VMT fractions for the various vehicle types, meteorological data, and vehicle fleet age mix. The fuel volatility is assumed to be a Reid vapor pressure of 9.0 psi. The VMT fractions for the various vehicle types were borrowed from the previous air quality analysis done in 2000 in which data was obtained from the Indiana Bureau of Motor Vehicles (BMV). The meteorological data, including average daily low and high temperatures (68 and 89 degrees Fahrenheit), sunrise and sunset hours (7AM and 9PM), and cloud cover (27%), was taken from the *Comparative Climatic Data Publication* by the National Climatic Data Center (NCDC) specifically for Evansville, Indiana, in the month of July. The national vehicle fleet age mix was used for heavy truck categories, but for passenger cars and light trucks the age distribution of household vehicles from the 2000 Evansville Household Travel Survey was used.

The thirty Mobile 6 runs differ in the VMT for which they estimate emissions rates. Each run calls a different set of the external data files created in the preprocessor. Fourteen

runs are made for freeways at various speeds, fourteen runs are made for arterials at various speeds, and one run each is made for local facilities and ramps. An example of the resulting Mobile 6 outputs are included in Appendix C.

Postprocessing of Mobile 6 Runs

The product of the Mobile 6 runs is a set of emissions rates. However, in order to determine conformity, these rates must be applied to the VMT to calculate the actual emissions. The post-processing of the Mobile 6 runs consists of this application of the Mobile 6 emissions rates and the generation of the report file.

Emissions are calculated differently for freeways and arterials than for local facilities and ramps. For freeways and arterials, emissions are computed for each link in the travel model's roadway network. Average link speeds by hour of the day (from POST_ALT) are used to distribute the hourly VMT on the link between the Mobile 6 speed bins using harmonic means as prescribed by Mobile 6 documentation. The emissions rates by facility type and speed bin are then applied to determine the emissions resulting from traffic on the link in each hour. The hourly emissions on each link are summed to produce the total emissions for the link and the emissions from each link are summed to produce total emissions for freeways and for arterials. The total daily emissions for each link are written to the link in the TransCAD network file in the fields: VOC, CO1, and NOX.

Since all local facilities and ramps are not included in the model's roadway network, their emissions cannot be calculated on a link-by-link basis. Moreover, Mobile 6 assumes specific speeds for these facility types based on national averages such that there would be little advantage to a link specific calculation in their case anyway. Therefore, emissions are computed for all local roads and all ramps with one calculation for each category in which the total VMT for the facility type (computed in the preprocessor) is multiplied by the emissions rates for the facility type.

Air Quality Conformity Analysis Report for Vanderburgh County from MOBILE6 and the Evansville Regional Travel Model Wed Oct 01 17:48:27 2003
Year: 2000
Scenario: Base Conditions in the Year 2000
4391345 VMT in Vanderburgh County
VOC CO NOX
Scenario: 9.01 tons/day 95.22 tons/day 12.44 tons/day
Budget: 16.29 tons/day 106.96 tons/day 12.52 tons/day

Figure 3: Air Quality Post-Processor Report

The total emissions for freeways and arterials are then added to the totals for local facilities and ramps to produce the grand total emissions for the analysis area. The program generates a report file in ASCII text format that reports the daily vehicle miles of travel (VMT) and daily emissions of volatile organic compounds (VOC), carbon monoxide (CO) and nitrogen oxides (NOX) for Vanderburgh County. The State Implementation Plan (SIP) budgets for the emissions of VOC, CO, and NOX are also included in the report to facilitate the determination of conformity. An example report can be seen above in Figure 3.

Appendix A: Mobile 6 Control Input File

The air quality post-processor produces a Mobile 6 control file which is used to specify the inputs for the thirty Mobile 6 model runs (or, scenarios, more precisely). This control file, named Mobile6.in, is created in and called from the TransCAD program's root directory. An example of a control file created and used by the air quality post-processor is presented below.

MOBILE6.in

MOBILE6 INPUT FILE :

*created by M6in macro written 6/13/03, vlb2

POLLUTANTS : HC CO NOX
REPORT FILE : C:\EUTS\Model_v3\post\M6REPORT.txt
SPREADSHEET : M6REPORT

RUN DATA
MIN/MAX TEMP : 68. 89.
FUEL RVP : 9.0

REG DIST : C:\EUTS\Model_v3\post\VREGDATA.D
VMT FRACTIONS :
0.617 0.044 0.148 0.059 0.027 0.032 0.003 0.002
0.002 0.007 0.008 0.009 0.033 0.002 0.001 0.006

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 0mph to 2.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_1.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt1.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 2.5mph to 7.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_2.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt2.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 7.5mph to 12.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_3.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt3.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 12.5mph to 17.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_4.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt4.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 17.5mph to 22.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_5.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt5.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 22.5mph to 27.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_6.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt6.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 27.5mph to 32.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_7.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt7.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 32.5mph to 37.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_8.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt8.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 37.5mph to 42.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_9.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt9.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 42.5mph to 47.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_10.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt10.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 47.5mph to 52.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_11.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt11.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 52.5mph to 57.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_12.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt12.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 57.5mph to 62.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_13.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt13.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Freeways 62.5mph to 500mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt1.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1_14.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt14.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 0mph to 2.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_1.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt1.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 2.5mph to 7.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_2.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt2.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 7.5mph to 12.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_3.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt3.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 12.5mph to 17.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_4.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt4.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 17.5mph to 22.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_5.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt5.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 22.5mph to 27.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_6.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt6.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 27.5mph to 32.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_7.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt7.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 32.5mph to 37.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_8.d
SPEED VMT : C:\EUTS\Model_v3\post\Svmt8.d

SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 37.5mph to 42.5mph
CALENDAR YEAR : 2000
EVALUATION MONTH : 7
CLOUD COVER : 0.27
SUNRISE/SUNSET : 7 9
VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_9.d

SPEED VMT : C:\EUTS\Model_v3\post\Svmt9.d
 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 42.5mph to 47.5mph
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_10.d
 SPEED VMT : C:\EUTS\Model_v3\post\Svmt10.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 47.5mph to 52.5mph
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_11.d
 SPEED VMT : C:\EUTS\Model_v3\post\Svmt11.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 52.5mph to 57.5mph
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_12.d
 SPEED VMT : C:\EUTS\Model_v3\post\Svmt12.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 57.5mph to 62.5mph
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_13.d
 SPEED VMT : C:\EUTS\Model_v3\post\Svmt13.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Arterials 62.5mph to 500mph
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt2.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt2_14.d
 SPEED VMT : C:\EUTS\Model_v3\post\Svmt14.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Locals
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt3.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1.d

 SCENARIO RECORD : Scenario Title : Vanderburg Co., 2000 Ramps
 CALENDAR YEAR : 2000
 EVALUATION MONTH : 7
 CLOUD COVER : 0.27
 SUNRISE/SUNSET : 7 9
 VMT BY FACILITY : C:\EUTS\Model_v3\post\Fvmt4.d
 VMT BY HOUR : C:\EUTS\Model_v3\post\Hvmt1.d

END OF RUN

Appendix B: Example Mobile 6 External Data Input Files

The air quality post-processor calls a number of external data files in its Mobile 6 model runs. In particular, it calls four types of files, the VREGDATA file, FVMT, SVMT, and HVMT files.

The VREGDATA.d file displayed below contains the distribution of vehicles by vehicle age for light duty vehicles. The distribution was taken from the 2000 Evansville Household Travel Survey and shows that there are slightly more newer vehicles in the Evansville area than assumed in the national defaults for Mobile 6. Due to a lack of local data the national defaults were used for heavy duty vehicles.

VREGDATA.d

```
REG DIST
*
* LDV
1 0.0746 0.0888 0.0782 0.0772 0.0600 0.0677 0.0749 0.0610 0.0477 0.0500
 0.0500 0.0494 0.0366 0.0333 0.0283 0.0216 0.0167 0.0144 0.0056 0.0061
 0.0061 0.0094 0.0089 0.0106 0.0229
* LDT1
2 0.0746 0.0888 0.0782 0.0772 0.0600 0.0677 0.0749 0.0610 0.0477 0.0500
 0.0500 0.0494 0.0366 0.0333 0.0283 0.0216 0.0167 0.0144 0.0056 0.0061
 0.0061 0.0094 0.0089 0.0106 0.0229
* LDT2
3 0.0746 0.0888 0.0782 0.0772 0.0600 0.0677 0.0749 0.0610 0.0477 0.0500
 0.0500 0.0494 0.0366 0.0333 0.0283 0.0216 0.0167 0.0144 0.0056 0.0061
 0.0061 0.0094 0.0089 0.0106 0.0229
* LDT3
4 0.0746 0.0888 0.0782 0.0772 0.0600 0.0677 0.0749 0.0610 0.0477 0.0500
 0.0500 0.0494 0.0366 0.0333 0.0283 0.0216 0.0167 0.0144 0.0056 0.0061
 0.0061 0.0094 0.0089 0.0106 0.0229
* LDT4
5 0.0746 0.0888 0.0782 0.0772 0.0600 0.0677 0.0749 0.0610 0.0477 0.0500
 0.0500 0.0494 0.0366 0.0333 0.0283 0.0216 0.0167 0.0144 0.0056 0.0061
 0.0061 0.0094 0.0089 0.0106 0.0229
```

The post-processor creates input files FVMT1.d through FVMT4.d and SVMT1.d through SVMT14.d but as these files only contain 1's and 0's to flag the appropriate facility type and speed bin and as they are of considerable length, they are not all reproduced here. Instead, one example of an SVMT file is given.

SVMT5.d

```
SPEED VMT
1 1 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 2 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 3 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 4 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 5 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 6 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 7 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 8 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 9 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 10 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 11 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
```

```

1 12 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 13 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 14 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 15 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 16 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 17 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 18 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 19 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 20 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 21 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 22 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 23 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
1 24 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 1 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 2 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 3 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 4 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 5 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 6 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 7 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 8 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 9 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 10 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 11 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 12 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 13 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 14 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 15 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 16 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 17 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 18 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 19 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 20 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 21 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 22 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 23 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
2 24 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

```

The post-processor also produces 29 HVMT files: HVMT1.d, HVMT1_1.d through HVMT1_14.d, and HVMT2_1.d through HVMT2_14.d. These files contain the hourly distribution of total daily VMT for each facility type and speed bin. HVMT1.d (shown below) is the overall average hourly distribution for the model used by the POST_ALT post-processor. HVMT1_1.d contains the average hourly distribution of VMT on freeways with speeds in speed bin 1 for that hour. HVMT2_14.d contains the average hourly distribution of VMT on arterials with speeds in speed bin 14 in that hour.

HVMT1.d

VMT BY HOUR

* Fraction of all vehicle miles traveled by hour of the day.

* First hour is 6 a.m.

*

0.0586	0.1042	0.0744	0.0504	0.0499	0.0629
0.0626	0.0549	0.0613	0.0834	0.0831	0.0807
0.0496	0.0320	0.0202	0.0171	0.0164	0.0090
0.0043	0.0025	0.0018	0.0018	0.0038	0.0151

Two additional examples of the hourly distribution of VMT on arterials with speeds in the third and eighth bins for that hour are presented below. These examples were based on using the average of the hourly distributions from the Evansville and Indiana household travel surveys which is contained in the file: EVIN24DTD.dbf. HVMT2_3.d shows that essentially all VMT on arterials in the third speed bin (7.5-12.5 mph) occurs as a result of congestion in the morning peak hour. HVMT2_8.d shows that the VMT on arterials in the eighth speed bin (32.5-37.5 mph) is more evenly distributed throughout the day with peaks from 6-9 AM and 3-6 PM.

HVMT2_3.d

VMT BY HOUR

* Fraction of all vehicle miles traveled by hour of the day.
* First hour is 6 a.m.
*
0.00001 0.99977 0.00001 0.00001 0.00001 0.00001
0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
0.00001 0.00001 0.00001 0.00001 0.00001 0.00001

HVMT2_8.d

VMT BY HOUR

* Fraction of all vehicle miles traveled by hour of the day.
* First hour is 6 a.m.
*
0.05342 0.09456 0.07875 0.04634 0.04585 0.05761
0.05735 0.05045 0.05619 0.09978 0.10017 0.09967
0.04556 0.02943 0.01865 0.01579 0.01508 0.00832
0.00394 0.00233 0.00169 0.00168 0.00350 0.01389

Ramps and Local facilities both use the HVMT1.d distribution reflecting the hourly distribution of all VMT in the model.

Appendix C: Mobile 6 Report Output File

An example of a Mobile6 report file created from the thirty model runs is presented below. The emissions rate table used by the post-processor and displayed in Figure 4 is created from the rates reported in the file reproduced below.

M6REPORT.TXT

```
*****
* MOBILE6.2.01 (31-Oct-2002) *
* Input file: MOBILE6.IN (file 1, run 1). *
*****  
  
* Reading Registration Distributions from the following external
* data file: C:\EUTS\MODEL_V3\POST\VREGDATA.D  
M615 Comment:  
    User supplied VMT mix.  
  
* # # # # # # # # # # # # # # # # # # # # # # # #  
* Scenario Title : Vanderburg Co., 2000 Freeways Omph to 2.5mph
* File 1, Run 1, Scenario 1.
* # # # # # # # # # # # # # # # # # # # # # # # #  
M617 Comment:  
    User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:  
    User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.  
  
* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D  
  
Reading User Supplied ROADWAY VMT Factors  
  
* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_1.D  
  
* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT1.D
M 48 Warning:  
    there are no sales for vehicle class HDGV8b  
  
    Calendar Year: 2000
    Month: July
    Altitude: Low
    Minimum Temperature: 68.0 (F)
    Maximum Temperature: 89.0 (F)
    Absolute Humidity: 75. grains/lb
    Nominal Fuel RVP: 9.0 psi
    Weathered RVP: 8.7 psi
    Fuel Sulfur Content: 300. ppm  
  
    Exhaust I/M Program: No
    Evap I/M Program: No
    ATP Program: No
    Reformulated Gas: No  
  
Vehicle Type: LDGV LDGT12 LDGT34 LDGT HDGV LDDV LDDT HDDV MC All Veh
  GVWR: <6000 >6000 (All)
----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----  
VMT Distribution: 0.6157 0.1918 0.0848 0.0301 0.0013 0.0013 0.0689 0.0060 1.0000  
-----  
Composite Emission Factors (g/mi):
  Composite VOC : 13.569 13.170 14.340 13.529 21.282 1.965 1.968 2.448 8.71 12.964
  Composite CO : 55.74 61.40 70.64 64.23 142.40 5.202 4.430 18.348 105.40 58.287
  Composite NOX : 2.311 2.492 2.745 2.570 3.852 2.910 2.604 30.607 1.07 4.372  
-----  
  
* # # # # # # # # # # # # # # # # # # # # # #  
* Scenario Title : Vanderburg Co., 2000 Freeways 2.5mph to 7.5mph
* File 1, Run 1, Scenario 2.
* # # # # # # # # # # # # # # # # # # # # # #  
M617 Comment:  
    User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:  
    User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.  
  
* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D  
  
Reading User Supplied ROADWAY VMT Factors  
  
* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_2.D  
  
* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT2.D
M 48 Warning:  
    there are no sales for vehicle class HDGV8b  
  
    Calendar Year: 2000
    Month: July
    Altitude: Low
    Minimum Temperature: 68.0 (F)
    Maximum Temperature: 89.0 (F)
```

Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
 Evap I/M Program: No
 ATP Program: No
 Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	5.421	5.669	6.195	5.830	10.511	1.772	1.767	2.151	5.89	5.455
Composite CO :	31.27	36.06	41.06	37.59	113.78	4.416	3.733	14.978	62.13	34.495
Composite NOX :	2.025	2.194	2.419	2.263	3.952	2.624	2.346	28.362	1.00	3.958

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderbilt Co., 2000 Freeways 7.5mph to 12.5mph
 * File 1, Run 1, Scenario 3.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT1_3.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT3.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
 Evap I/M Program: No
 ATP Program: No
 Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	3.013	3.211	3.491	3.296	6.391	1.471	1.454	1.689	3.75	3.102
Composite CO :	19.37	23.42	26.29	24.30	75.70	3.331	2.772	10.328	31.57	21.835
Composite NOX :	1.449	1.574	1.747	1.627	4.152	2.193	1.957	24.965	0.94	3.198

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderbilt Co., 2000 Freeways 12.5mph to 17.5mph
 * File 1, Run 1, Scenario 4.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT1_4.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT4.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Vehicle Type: LDGV LDGT12 LDGT34 LDGT HDGV LDDV LDDT HDDV MC All Veh
 GVWR: <6000 >6000 (All)

VMT Distribution:	0.6157	0.1918	0.0848	0.0301	0.0013	0.0013	0.0689	0.0060	1.0000
-------------------	--------	--------	--------	--------	--------	--------	--------	--------	--------

Composite Emission Factors (g/mi):

Composite VOC :	1.954	2.035	2.215	2.090	3.119	0.979	0.941	0.932	2.47	1.957
-----------------	-------	-------	-------	-------	-------	-------	-------	-------	------	-------

Composite CO :	15.64	19.38	21.65	20.08	31.01	1.959	1.557	4.449	14.32	16.515
----------------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------

Composite NOX :	1.188	1.304	1.450	1.349	4.753	1.575	1.401	20.111	1.07	2.644
-----------------	-------	-------	-------	-------	-------	-------	-------	--------	------	-------

* #

* Scenario Title : Vanderburg Co., 2000 Freeways 27.5mph to 32.5mph

* File 1, Run 1, Scenario 7.

* #

M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_7.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT7.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							

VMT Distribution:	0.6157	0.1918	0.0848	0.0301	0.0013	0.0013	0.0689	0.0060	1.0000
-------------------	--------	--------	--------	--------	--------	--------	--------	--------	--------

Composite Emission Factors (g/mi):

Composite VOC :	1.861	1.948	2.117	1.999	2.720	0.893	0.851	0.799	2.32	1.852
-----------------	-------	-------	-------	-------	-------	-------	-------	-------	------	-------

Composite CO :	15.50	19.26	21.49	19.94	25.71	1.779	1.398	3.680	12.30	16.163
----------------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------

Composite NOX :	1.189	1.310	1.456	1.355	4.954	1.510	1.342	19.599	1.13	2.617
-----------------	-------	-------	-------	-------	-------	-------	-------	--------	------	-------

* #

* Scenario Title : Vanderburg Co., 2000 Freeways 32.5mph to 37.5mph

* File 1, Run 1, Scenario 8.

* #

M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_8.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT8.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							

VMT Distribution:	0.6157	0.1918	0.0848	0.0301	0.0013	0.0013	0.0689	0.0060	1.0000
-------------------	--------	--------	--------	--------	--------	--------	--------	--------	--------

Composite Emission Factors (g/mi):

Composite VOC :	1.777	1.862	2.017	1.909	2.448	0.829	0.784	0.700	2.20	1.759
Composite CO :	15.67	19.49	21.70	20.17	22.52	1.664	1.296	3.185	10.80	16.194
Composite NOX :	1.184	1.309	1.454	1.353	5.154	1.497	1.331	19.496	1.18	2.612

* # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Freeways 37.5mph to 42.5mph
* File 1, Run 1, Scenario 9.
* # # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_9.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT9.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.730	1.818	1.967	1.864	2.260	0.781	0.735	0.627	2.11	1.707
Composite CO :	16.42	20.35	22.57	21.03	20.84	1.594	1.234	2.886	9.76	16.818
Composite NOX :	1.196	1.325	1.469	1.369	5.354	1.535	1.364	19.791	1.21	2.651

* # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Freeways 42.5mph to 47.5mph
* File 1, Run 1, Scenario 10.
* # # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_10.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT10.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.689	1.779	1.921	1.823	2.127	0.747	0.699	0.575	2.06	1.662
Composite CO :	17.18	21.20	23.44	21.89	20.37	1.559	1.203	2.736	9.10	17.492
Composite NOX :	1.213	1.348	1.492	1.392	5.555	1.627	1.447	20.515	1.23	2.724

M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_13.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT13.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.660	1.740	1.865	1.778	2.015	0.701	0.651	0.504	2.36	1.625
Composite CO :	22.22	25.94	28.25	26.65	29.07	1.636	1.271	3.064	17.14	22.250
Composite NOX :	1.299	1.446	1.588	1.490	6.083	2.379	2.125	26.432	1.53	3.231

* # # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Freeways 62.5mph to 500mph
* File 1, Run 1, Scenario 14.
* # # # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT1.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1_14.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT14.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.625	1.698	1.817	1.735	1.994	0.701	0.651	0.504	2.65	1.593
Composite CO :	22.53	26.42	28.74	27.13	34.64	1.734	1.358	3.485	24.10	22.809
Composite NOX :	1.309	1.463	1.605	1.506	6.293	2.897	2.592	30.503	1.67	3.531

* # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Arterials 0mph to 2.5mph
* File 1, Run 1, Scenario 15.
* # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D
 Reading User Supplied ROADWAY VMT Factors
 * Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT2_1.D
 * Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT1.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm

Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):										
Composite VOC :	13.569	13.170	14.341	13.529	21.282	1.965	1.968	2.448	8.71	12.964
Composite CO :	55.74	61.40	70.64	64.23	142.40	5.202	4.430	18.348	105.40	58.287
Composite NOX :	2.311	2.492	2.745	2.570	3.852	2.910	2.604	25.709	1.07	4.035

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderburg Co., 2000 Arterials 2.5mph to 7.5mph
 * File 1, Run 1, Scenario 16.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D
 Reading User Supplied ROADWAY VMT Factors
 * Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT2_2.D
 * Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT2.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm

Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):										
Composite VOC :	5.018	5.396	5.908	5.553	9.963	1.772	1.767	2.151	5.84	5.114
Composite CO :	29.36	36.08	41.00	37.59	108.83	4.416	3.733	14.978	57.79	33.140
Composite NOX :	1.885	2.152	2.378	2.221	3.997	2.624	2.346	23.464	1.03	3.524

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderburg Co., 2000 Arterials 7.5mph to 12.5mph
 * File 1, Run 1, Scenario 17.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D
 Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT2_3.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT3.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm

Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWVR:	<6000	>6000								
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	2.927	3.220	3.497	3.305	6.154	1.471	1.454	1.689	3.72	3.044
Composite CO :	19.43	24.65	27.68	25.58	72.41	3.331	2.772	10.328	29.49	22.115
Composite NOX :	1.572	1.784	1.976	1.843	4.200	2.193	1.957	20.067	0.97	2.998

* # # # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Arterials 12.5mph to 17.5mph
* File 1, Run 1, Scenario 18.
* # # # # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT2_4.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT4.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm

Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWVR:	<6000	>6000								
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	2.375	2.559	2.779	2.626	4.576	1.254	1.228	1.355	3.00	2.441
Composite CO :	16.91	21.57	24.21	22.38	51.05	2.660	2.178	7.453	20.39	18.781
Composite NOX :	1.379	1.554	1.724	1.606	4.401	1.897	1.691	17.744	0.98	2.659

* # # # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Arterials 17.5mph to 22.5mph
* File 1, Run 1, Scenario 19.
* # # # # # # # # # # # # # # # # # # #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT2_5.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT5.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm
Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	2.079	2.188	2.380	2.247	3.629	1.096	1.063	1.112	2.67	2.107
Composite CO :	15.78	20.11	22.56	20.86	38.29	2.234	1.801	5.628	16.35	17.133
Composite NOX :	1.287	1.441	1.600	1.490	4.597	1.700	1.513	16.190	1.03	2.469

Scenario Title : Vanderburg Co., 2000 Arterials 22.5mph to 27.5mph
File 1, Run 1, Scenario 20.

M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

Reading Hourly Roadway VMT distribution from the following external data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

Reading Hourly VMT distribution from the following external data file: C:\EUTS\MODEL_V3\POST\HVMT2_6.D

Reading Hourly, Roadway, and Speed VMT dist. from the following external data file: C:\EUTS\MODEL_V3\POST\SVMT6.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July
Altitude:	Low
Minimum Temperature:	68.0 (F)
Maximum Temperature:	89.0 (F)
Absolute Humidity:	75. grains/lb
Nominal Fuel RVP:	9.0 psi
Weathered RVP:	8.7 psi
Fuel Sulfur Content:	300. ppm
Exhaust I/M Program:	No
Evap I/M Program:	No
ATP Program:	No
Reformulated Gas:	No

Vehicle Type:	LDGV	LDGT12 <6000	LDGT34 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	2.050	2.114	2.295	2.170	3.237	0.979	0.941	0.932	2.49	2.042
Composite CO :	16.93	20.43	22.78	21.15	33.18	1.959	1.557	4.449	15.32	17.677
Composite NOX :	1.292	1.405	1.556	1.451	4.729	1.575	1.401	15.213	1.06	2.398

Scenario Title : Vanderburg Co., 2000 Arterials 27.5mph to 32.5mph
File 1, Run 1, Scenario 21.

M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

Reading Hourly Roadway VMT distribution from the following external data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

Reading Hourly VMT distribution from the following external data file: C:\EUTS\MODEL_V3\POST\HVMT2_7.D

Reading Hourly, Roadway, and Speed VMT dist. from the following external data file: C:\EUTS\MODEL_V3\POST\SVMT7.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year:	2000
Month:	July

Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWVR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.964	2.030	2.202	2.082	2.865	0.893	0.851	0.799	2.34	1.943
Composite CO :	17.23	20.51	22.78	21.21	28.13	1.779	1.398	3.680	13.48	17.665
Composite NOX :	1.259	1.361	1.506	1.405	4.901	1.510	1.342	14.701	1.10	2.335

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderburg Co., 2000 Arterials 32.5mph to 37.5mph
 * File 1, Run 1, Scenario 22.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT2_8.D
 * Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT8.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
 Evap I/M Program: No
 ATP Program: No
 Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWVR:	<6000	>6000	(All)							
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.861	1.933	2.092	1.982	2.570	0.829	0.784	0.700	2.21	1.835
Composite CO :	17.44	20.79	23.01	21.47	24.54	1.664	1.296	3.185	11.75	17.708
Composite NOX :	1.229	1.337	1.482	1.382	5.097	1.497	1.331	14.598	1.14	2.308

* # # # # # # # # # # # # # # # # # #
 * Scenario Title : Vanderburg Co., 2000 Arterials 37.5mph to 42.5mph
 * File 1, Run 1, Scenario 23.
 * # # # # # # # # # # # # # # # # # #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT2_9.D
 * Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT9.D
 M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi

Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type: GVWR:	LDGV <6000	LDGT12 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh	
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000
<hr/>										
Composite Emission Factors (g/mi):										
Composite VOC :	1.802	1.882	2.034	1.928	2.360	0.781	0.735	0.627	2.12	1.772
Composite CO :	18.24	21.70	23.94	22.39	22.59	1.594	1.234	2.886	10.52	18.370
Composite NOX :	1.234	1.351	1.494	1.395	5.299	1.535	1.364	14.893	1.17	2.342

* #
* Scenario Title : Vanderburg Co., 2000 Arterials 42.5mph to 47.5mph
* File 1, Run 1, Scenario 24.
* #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT2_10.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT10.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No
ATP Program: No
Reformulated Gas: No

Vehicle Type: GVWR:	LDGV <6000	LDGT12 >6000	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh	
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000
<hr/>										

Composite Emission Factors (g/mi):
 Composite VOC : 1.752 1.839 1.985 1.884 2.213 0.747 0.699 0.575 2.07 1.721
 Composite CO : 19.10 22.66 24.92 23.35 22.05 1.559 1.203 2.736 9.77 19.137
 Composite NOX : 1.247 1.372 1.515 1.416 5.499 1.627 1.447 15.617 1.19 2.412

* #
* Scenario Title : Vanderburg Co., 2000 Arterials 47.5mph to 52.5mph
* File 1, Run 1, Scenario 25.
* #
M617 Comment:
User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:
User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT2_11.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
* data file: C:\EUTS\MODEL_V3\POST\SVMT11.D

M 48 Warning:
there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
Evap I/M Program: No

ATP Program: No
Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000								
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.711	1.800	1.938	1.842	2.109	0.723	0.674	0.538	2.06	1.678
Composite CO :	20.02	23.65	25.92	24.34	22.77	1.554	1.199	2.715	9.47	19.992
Composite NOX :	1.262	1.394	1.537	1.438	5.696	1.784	1.589	16.853	1.26	2.519

* #

* Scenario Title : Vanderburg Co., 2000 Arterials 52.5mph to 57.5mph

* File 1, Run 1, Scenario 26.

* #

M617 Comment:

User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:

User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external

* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external

* data file: C:\EUTS\MODEL_V3\POST\HVMT2_12.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external

* data file: C:\EUTS\MODEL_V3\POST\SVMT12.D

M 48 Warning:

there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No

Evap I/M Program: No

ATP Program: No

Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000								
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):

Composite VOC :	1.680	1.765	1.896	1.805	2.043	0.708	0.659	0.515	2.06	1.645
Composite CO :	21.02	24.68	26.96	25.38	24.88	1.579	1.220	2.820	9.49	20.967
Composite NOX :	1.278	1.417	1.559	1.460	5.889	2.024	1.806	18.745	1.40	2.673

* #

* Scenario Title : Vanderburg Co., 2000 Arterials 57.5mph to 62.5mph

* File 1, Run 1, Scenario 27.

* #

M617 Comment:

User supplied alternate AC input: Cloud Cover Fraction set to 0.27.

M618 Comment:

User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external

* data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external

* data file: C:\EUTS\MODEL_V3\POST\HVMT2_13.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external

* data file: C:\EUTS\MODEL_V3\POST\SVMT13.D

M 48 Warning:

there are no sales for vehicle class HDGV8b

Calendar Year: 2000
Month: July
Altitude: Low
Minimum Temperature: 68.0 (F)
Maximum Temperature: 89.0 (F)
Absolute Humidity: 75. grains/lb
Nominal Fuel RVP: 9.0 psi
Weathered RVP: 8.7 psi
Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No

Evap I/M Program: No

ATP Program: No

Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT (All)	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GVWR:	<6000	>6000								

VMT Distribution: 0.6157 0.1918 0.0848 0.0301 0.0013 0.0013 0.0689 0.0060 0.0060 1.0000

Composite Emission Factors (g/mi):
 Composite VOC : 1.649 1.728 1.853 1.767 2.001 0.701 0.651 0.504 2.36 1.615
 Composite CO : 21.74 25.53 27.83 26.24 28.53 1.636 1.271 3.064 16.82 21.820
 Composite NOX : 1.293 1.440 1.582 1.483 6.093 2.379 2.125 21.535 1.54 2.889

* #
 * Scenario Title : Vanderburg Co., 2000 Arterials 62.5mph to 500mph
 * File 1, Run 1, Scenario 28.
 * #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT2.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT2_14.D

* Reading Hourly, Roadway, and Speed VMT dist. from the following external
 * data file: C:\EUTS\MODEL_V3\POST\SVMT14.D

M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
 Evap I/M Program: No
 ATP Program: No
 Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWWR:		<6000	>6000	(All)						
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):
 Composite VOC : 1.622 1.695 1.814 1.731 1.990 0.701 0.651 0.504 2.65 1.590
 Composite CO : 22.36 26.26 28.57 26.97 34.41 1.734 1.358 3.485 23.95 22.653
 Composite NOX : 1.307 1.460 1.602 1.503 6.295 2.897 2.592 25.605 1.67 3.191

* #
 * Scenario Title : Vanderburg Co., 2000 Locals
 * File 1, Run 1, Scenario 29.
 * #
 M617 Comment:
 User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
 M618 Comment:
 User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\FVMT3.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
 * data file: C:\EUTS\MODEL_V3\POST\HVMT1.D

M 48 Warning:
 there are no sales for vehicle class HDGV8b

Calendar Year: 2000
 Month: July
 Altitude: Low
 Minimum Temperature: 68.0 (F)
 Maximum Temperature: 89.0 (F)
 Absolute Humidity: 75. grains/lb
 Nominal Fuel RVP: 9.0 psi
 Weathered RVP: 8.7 psi
 Fuel Sulfur Content: 300. ppm

Exhaust I/M Program: No
 Evap I/M Program: No
 ATP Program: No
 Reformulated Gas: No

Vehicle Type:	LDGV	LDGT12	LDGT34	LDGT	HDGV	LDDV	LDDT	HDDV	MC	All Veh
GWWR:		<6000	>6000	(All)						
VMT Distribution:	0.6157	0.1918	0.0848		0.0301	0.0013	0.0013	0.0689	0.0060	1.0000

Composite Emission Factors (g/mi):
 Composite VOC : 2.838 2.892 3.132 2.966 5.633 1.337 1.314 1.482 3.28 2.863
 Composite CO : 15.36 19.56 22.14 20.35 66.33 2.904 2.394 8.500 27.28 17.841
 Composite NOX : 1.241 1.292 1.439 1.337 4.225 2.008 1.790 15.315 0.92 2.327

```

* # # # # # # # # # # # # # # # # # #
* Scenario Title : Vanderburg Co., 2000 Ramps
* File 1, Run 1, Scenario 30.
* # # # # # # # # # # # # # # # # # #
M617 Comment:
    User supplied alternate AC input: Cloud Cover Fraction set to 0.27.
M618 Comment:
    User supplied alternate AC input: Sunrise at 7 AM, Sunset at 9 PM.

* Reading Hourly Roadway VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\FVMT4.D

Reading User Supplied ROADWAY VMT Factors

* Reading Hourly VMT distribution from the following external
* data file: C:\EUTS\MODEL_V3\POST\HVMT1.D
M 48 Warning:
    there are no sales for vehicle class HDGV8b

    Calendar Year: 2000
    Month: July
    Altitude: Low
    Minimum Temperature: 68.0 (F)
    Maximum Temperature: 89.0 (F)
    Absolute Humidity: 75. grains/lb
    Nominal Fuel RVP: 9.0 psi
    Weathered RVP: 8.7 psi
    Fuel Sulfur Content: 300. ppm

    Exhaust I/M Program: No
    Evap I/M Program: No
    ATP Program: No
    Reformulated Gas: No

Vehicle Type: LDGV LDGT12 LDGT34 LDGT HDGV LDDV LDDT HDDV MC All Veh
GVWR: <6000 >6000 (All)
-----
VMT Distribution: 0.6157 0.1918 0.0848 0.0301 0.0013 0.0013 0.0689 0.0060 1.0000
-----
Composite Emission Factors (g/mi):
    Composite VOC : 2.124 2.209 2.362 2.256 2.579 0.833 0.789 0.707 2.21 2.074
    Composite CO : 29.59 30.55 32.84 31.25 24.58 1.671 1.302 3.217 11.70 27.898
    Composite NOX : 1.471 1.659 1.812 1.706 5.085 1.496 1.330 11.070 1.14 2.304
-----
```

Appendix D: Associated GIS/DK Macros Code

The GIS/DK resource code for the EUTS model's POST_ALT and air quality post-processor macros are reproduced here from version 3.5 of the model.

```
Macro "getnames" (in_value)
    nnpath = in_value[1]

//Reads in the fields on the link layer from the netnames.dbf file

    nnfile = nnpath + "\\Netnames.dbf"
    netnames = OpenTable("netnames", "DBASE", (nnfile,))

    SetView (netnames)
    rec = GetFirstRecord(netnames + "|",null)
    while rec <> null do

        Var = netnames.VAR
        Name = netnames.NAME

        If Var = "ID" then ID = Name
        If Var = "LENGTH" then LENGTH = Name
        If Var = "FC" then FC = Name
        If Var = "LANES" then LANES = Name
        If Var = "AMAUTO" then AMAUTO = Name
        If Var = "PMAUTO" then PMAUTO = Name
        If Var = "OPAUTO" then OPAUTO = Name
        If Var = "AMTRK" then AMTRK = Name
        If Var = "PMTRK" then PMTRK = Name
        If Var = "OPTRK" then Optrk = Name
        If Var = "PKCAP" then PKCAP = Name
        If Var = "BPRA" then BPRA = Name
        If Var = "BPRB" then BPRB = Name
        If Var = "FFTMIN" then FFTMIN = Name
        If Var = "MDT" then MDT = Name
        If Var = "MDAT" then MDAT = Name
        If Var = "MDTT" then MDTT = Name
        If Var = "ADJFFT" then ADJFFT = Name
        If Var = "MIDAUTO" then MIDAUTO = Name
        If Var = "MIDTRK" then MIDTRK = Name
        If Var = "AREA" then AREA = Name

        If Var = "US41ID" then US41ID = Name
        If Var = "TBID" then BridgeID = Name
        If Var = "COR" then COR = Name
        If Var = "M6FT" then M6FT = Name
        If Var = "CO" then CO = Name
        If Var = "RTENM" then RTENM = Name
        If Var = "FTFC" then FTFC = Name
        If Var = "ABPKCAP" then ABPKCAP = Name
        If Var = "BAPKCAP" then BAPKCAP = Name

    rec = GetNextRecord(netnames + "|",null,null)
end

CloseView(netnames)

    thefields = {ID, LENGTH, FC, LANES, AMAUTO, PMAUTO, MIDAUTO, OPAUTO,
                AMTRK, PMTRK, MIDTRK, Optrk,
                PKCAP, BPRA, BPRB, FFTMIN, MDT, MDAT, MDTT, ADJFFT, AREA,
                US41ID, BridgeID, COR, M6FT, CO, RTENM, FTFC, ABPKCAP, BAPKCAP}

    Return(thefields)

endMacro

// =====

Macro "postalt" (in_value)
    thepath = in_value[1]
    linevw = in_value[2]
    dsplit = in_value[3]
    todass = in_value[4]

/*
Created by Vincent Bernardin, Jr.
of Bernardin, Lochmueler and Assoc.
for the Muncie, IN model
using the hourly distribution from the EUTS HH travel survey
04/25/03

refined and updated by the author for Lexington, KY
incorporating the half-hourly distribution from the Knoxville, TN
HH travel survey, allowing for user-defined distributions from
external data files, and adding statistics for area type
05/08/03

refined and updated by the same for Evansville, IN to include
hourly speeds and volumes for the Mobile6 macro 4M6 and adding
statistics for calibration corridors
06/10/03

refined and updated again by the author for the US 31 Plymouth-
South Bend EIS to calculate and report accident statistics, report
statistics on the US 31 corridor, to exclude links outside

```

Marshall, St. Joseph, and Elkhart counties, to report statistics
by county, and to reflect new LOS v/c breakpoints from HCM 2000
07/01/03

```

converted back for to include the new accident statistics, county
breakout, and LOS breakpoints in the EUTS version
07/02/03
*/
//Get network link layer

//    linevw = GetLayer()
//    ShowMessage("The current layer is " + linevw)
//    netfilename = GetLayerDB(linevw)

//Get path to modeling directory
//    thepath = ChooseDirectory("Choose the Modeling Directory", )

//Ns is the number of statistics (records) in the report file
Ns = 43

//Choose output file name

links = GetRecordCount(linevw, )

opt = {{"Initial Directory", thepath + "\\post\\"}, 
       {"Suggested Name", "post_rep"}, 
       outfile = ChooseFileName({{"dBASE", "*.*dbf"}}, "Choose Report File", opt)

linkfile = "Post_Alt Link File"
linkpath = SplitPath(outfile)
linkfilename = thepath + "\\post\\" + linkpath[3] + ".links.dbf"
mfilename = thepath + "\\post\\" + linkpath[3] + ".24M6.dbf"
SaveArray({linkfilename, mfilename}, thepath + "\\post\\post.gar")

//Read in the Network Field Names from the auxillary file
thefields = RunMacro("getnames", {thepath + "\\post\\"})

//Get daily distribution from auxillary file

errno = 0
getfiles:
dtdfile = ChooseFile({{"Daily Trip Distribution File", "*DTD.dbf"}}, 
                      "Choose the daily trip distribution file",
                      {"Initial Directory", thepath + "\\post\\"}, {})

dtdvw = OpenTable("dtdview", "dBASE", {dtdfile, })
dtdrecs = GetRecordCount(dtdvw, )
dim DAILY100[dtdrecs]
i = 0
rec = GetFirstRecord(dtdvw + "|", )
while rec <> null do
i = i + 1
DAILY100[i] = dtdvw.DTD
rec = GetNextRecord(dtdvw + "|", , )
end
CloseView(dtdvw)

//Get TOD periods from auxillary file

//    todfile = ChooseFile({{"Time of Day Periods File", "*TOD.dbf"}}, 
//                          "Choose the time of day periods file",
//                          {"Initial Directory", thepath + "\\post\\"}, {})

todfile = thepath + "\\post\\Eville24TOD.dbf"

todvw = OpenTable("todview", "dBASE", {todfile, })
todrecs = GetRecordCount(todvw, )
dim PERIODS[todrecs]
i = 0
rec = GetFirstRecord(todvw + "|", )
while rec <> null do
i = i + 1
PERIODS[i] = todvw.PERIOD
rec = GetNextRecord(todvw + "|", , )
end
CloseView(todvw)

//Check for inconsistencies

if dtdrecs <> todrecs then do
    ShowMessage("Error: The distribution and periods are inconsistent!")
    errno = errno + 1
    if errno < 4 then goto getfiles
    if errno = 4 then goto quit
end
dp = dtdrecs

EnableProgressBar("Post_Alt", 1)
CreateProgressBar("loading...", "False")

//Initialize some variables

RM = 0
RLM = 0
Tot_Delay = 0
MAXMAXVC = 0
TOTVC = 0

TOTVHT = 0

```

```

TOTVMT = 0
TOTVHT_Auto = 0
TOTVHT_Trk = 0
TOTVMT_Auto = 0
TOTVMT_Trk = 0
TOTFX = 0
TOTIX = 0
TOTPX = 0
TOTTX = 0

ESPI_VMT = 0
ESPI_VHT = 0
ESPI_VMT_CAR = 0
ESPI_VHT_CAR = 0
ESPI_VMT_TRK = 0
ESPI_VHT_TRK = 0

//Remember: if you change CLEVELS you may need to change the ESPI calculations
//report variables may also need to be adjusted
CLEVELS = {0.50, 0.70, 0.84, 0.99}
Ls = CLEVELS.length
dim VMTVC[Ls]
dim VHHTVC[Ls]
dim AMTVC[Ls]
dim AHTVC[Ls]
dim TMTVC[Ls]
dim THHTVC[Ls]
dim CLANEMI[Ls]
For j = 1 to Ls do
    VMTVC[j] = 0
    VHHTVC[j] = 0
    AMTVC[j] = 0
    AHTVC[j] = 0
    TMTVC[j] = 0
    THHTVC[j] = 0
    CLANEMI[j] = 0
end
FCLASS = {1, 2, 6, 7, 8, 9, 11, 12, 14, 16, 17, 19}
dim CVC[Ns]
dim RDMILES[Ns]
dim VMTC[Ns]
dim VHTC[Ns]
dim VMTAUTO[Ns]
dim VHHTAUTO[Ns]
dim VMTTRK[Ns]
dim VHHTTRK[Ns]
dim SPD[CNs]
dim SUMFX[Ns]
dim SUMIX[Ns]
dim SUMPX[Ns]
dim SUMTX[Ns]
for i = 1 to FCLASS.length do
    CVC[i] = 0
    RDMILES[i] = 0
    VMTC[i] = 0
    VHTC[i] = 0
    VMTAUTO[i] = 0
    VHHTAUTO[i] = 0
    VMTTRK[i] = 0
    VHHTTRK[i] = 0
    SUMFX[i] = 0
    SUMIX[i] = 0
    SUMPX[i] = 0
    SUMTX[i] = 0
end
AREA = {"CBD", "URB", "SUB", "RUR"}
COS = {"26", "KY", "65", "82", "87"}
CORRS = {1, 2, 3, 4, 5, 6, 7}
lc = 16 + AREA.length
for i = 17 to lc do
    CVC[i] = 0
    RDMILES[i] = 0
    VMTC[i] = 0
    VHTC[i] = 0
    VMTAUTO[i] = 0
    VHHTAUTO[i] = 0
    VMTTRK[i] = 0
    VHHTTRK[i] = 0
    SUMFX[i] = 0
    SUMIX[i] = 0
    SUMPX[i] = 0
    SUMTX[i] = 0
end
lc2 = lc + 2 + COS.length
fc2 = lc + 3
for i = fc2 to lc2 do
    CVC[i] = 0
    RDMILES[i] = 0
    VMTC[i] = 0
    VHTC[i] = 0
    VMTAUTO[i] = 0
    VHHTAUTO[i] = 0
    VMTTRK[i] = 0
    VHHTTRK[i] = 0
    SUMFX[i] = 0
    SUMIX[i] = 0
    SUMPX[i] = 0
    SUMTX[i] = 0
end
lc3 = lc2 + 2 + CORRS.length
fc3 = lc2 + 3

```

```

for i = fc3 to lc3 do
    CVC[i] = 0
    RDIMILES[i] = 0
    VMTC[i] = 0
    VHTC[i] = 0
    VMTAUTO[i] = 0
    VHTAUTO[i] = 0
    VMTRK[i] = 0
    VHTTRK[i] = 0
    SUMFX[i] = 0
    SUMIX[i] = 0
    SUMPX[i] = 0
    SUMTX[i] = 0
end

VHT_US41 = 0
VHTTRK_US41 = 0
VHTAUTO_US41 = 0
VMT_US41 = 0
VMTRK_US41 = 0
VMTAUTO_US41 = 0

//The following accident rates by functional class are based on
//Indiana's statistics for 1997-1999

FXR = {0.73, 1.96, x, x, x, 2.06, 1.38, 1.97, 3.65,
       x, 0.33, 3.12, x, 0.61, x, 0.93, 1.56, x, 0.68}
IXR = {17.53, 43.15, x, x, x, 53.48, 81.13, 89.67, 87.96,
       x, 14.50, 45.27, x, 99.50, x, 110.51, 113.24, x, 111.89}
PXR = {68.08, 137.45, x, x, x, 155.98, 232.66, 254.57, 249.45,
       x, 55.38, 132.28, x, 304.39, x, 365.16, 376.64, x, 372.67}
TXR = {86.34, 186.57, x, x, x, 211.52, 315.18, 346.21, 341.07,
       x, 70.21, 180.67, x, 404.50, x, 476.59, 491.43, x, 485.24}

//Daily distribution of traffic and period directional factor

np = 1 + R2I(ArrayMax(PERIODS))
dim persum[np]
for i = 1 to np do
    persum[i] = 0
end
for i = 1 to np do
    for j = 1 to dp do
        if PERIODS[j] = i - 1 then do
            persum[i] = persum[i] + DAILY100[j]
        end
    end
end
dim DDIST[dp]
for i = 1 to dp do
    tp = 1 + PERIODS[i]
    if todass = 0 then DDIST[i] = DAILY100[i]
    if todass = 1 then DDIST[i] = DAILY100[i] / persum[tp]
end
dim PDIR[dp]
for i = 1 to dp do
    PDIR[i] = 0.5
    if PERIODS[i] > 0 then PDIR[i] = dsplit/100
end

//Initialize daily period volume arrays

dim PCEVOL[dp]
dim ADTVOL[dp]
dim CARVOL[dp]
dim TRKVOL[dp]

//Create the output files

lfile = CreateTable(linkfile, linkfilename, "dBASE", {{"LID", "Integer", 16, null, "No"}, {"FHWA_FC", "Integer", 8, null, "No"}, {"MAXVCL", "Real", 8, 3, "No"}, {"AVGSP", "Real", 8, 2, "No"}, {"AVGTT1", "Real", 6, 3, "No"}, {"WORSTLOS", "String", 1, null, "No"}, {"HRS_DELAY", "Real", 16, 3, "No"}, {"VHT", "Real", 10, 3, "No"}, {"VMTT", "Real", 12, 3, "No"}, {"BPERA", "Real", 12, 3, "No"}, {"BPRB", "Real", 12, 3, "No"}, {"FFTME", "Real", 12, 3, "No"}, {"BAD", "Integer", 16, null, "No"}, {"AMAVGTT", "Real", 12, 3, "No"}, {"PMAVGTT", "Real", 12, 3, "No"}, {"OFAVGTT", "Real", 12, 3, "No"}, {"FX", "Real", 12, 3, "No"}, {"IX", "Real", 12, 3, "No"}, {"PX", "Real", 12, 3, "No"}, {"TX", "Real", 12, 3, "No"}})

dim carvolfield[24]
dim trkvolfield[24]
dim avgspdfield[24]
dim mfields[73]
mfields[1] = {"LID", "Integer", 16, null, "No"}
for i = 1 to 24 do
    if i > 9 then do
        carvolfield[i] = "CARVOL_" + i2s(i)
        trkvolfield[i] = "TRKVOL_" + i2s(i)
        avgspdfield[i] = "AVGSPD_" + i2s(i)
    end
    else do

```

```

        carvolfield[i] = "CARVOL_0" + i2s(i)
        trkvolfield[i] = "TRKVOL_0" + i2s(i)
        avgspdfield[i] = "AVGSPD_0" + i2s(i)
    end
    a = i + 1
    b = i + 25
    c = i + 49
    mfields[a] = {carvolfield[i], "Real", 12, 3, "No"}
    mfields[b] = {trkvolfield[i], "Real", 12, 3, "No"}
    mfields[c] = {avgspdfield[i], "Real", 12, 3, "No"}
end

mfile = CreateTable("Mobile6 Summary", mfilename, "dBASE", mfields)

postrep = CreateTable("Post_Alt Report", outfilename, "dBASE",
    {"SystemStat", "String", 30, null, "No"}, {"Stat", "String", 20, null, "No"}, {"Class", "String", 45, null, "No"}, {"FHWA_FC", "Integer", 10, null, "No"}, {"RoadMiles", "Real", 8, 2, "No"}, {"VMT", "Integer", 10, null, "No"}, {"Auto_VMT", "Integer", 10, null, "No"}, {"Truck_VMT", "Integer", 10, null, "No"}, {"VHT", "Integer", 10, null, "No"}, {"Auto_VHT", "Integer", 10, null, "No"}, {"Truck_VHT", "Integer", 10, null, "No"}, {"AvgSpeed", "Real", 12, 2, "No"}, {"VC", "Real", 12, 2, "No"}, {"Fatal_Xs", "Integer", 10, null, "No"}, {"Injury_Xs", "Integer", 10, null, "No"}, {"PDO_Xs", "Integer", 10, null, "No"}, {"Total_Xs", "Integer", 10, null, "No"}}

//Loop through the assigned network

SetView(linevw)
recno = 0
rec = GetFirstRecord(linevw + "|", null)
while rec <> null do

    recno = recno + 1
    prog = Round ((recno/links)*98, 0)
    UpdateProgressBar("Computing Statistics", prog)
    SetStatus(1, "Record " + i2s(recno) + " of " + i2s(links), )

//Read in values for the record from the fields on the link layer

//      thefields = {ID, LENGTH, FC, LANES, AMAUTO, PMAUTO, MIDAUTO, OPAUTO,
//                     AMTRK, PMTRK, MIDTRK, OPTRK,
//                     PKCAP, BPRA, BPRB, FFTMIN, MDT, MDAT, MDTT, ADJFFT, AREA,
//                     US41ID, BridgeID, COR}
//
//      ID = linevw.(thefields[1])
//      LENGTH = linevw.(thefields[2])
//      FC = linevw.(thefields[3])
//      LANES = linevw.(thefields[4])
//      AMAUTO = linevw.(thefields[5])
//      PMAUTO = linevw.(thefields[6])
//      OPAUTO = linevw.(thefields[8])
//      AMTRK = linevw.(thefields[9])
//      PMTRK = linevw.(thefields[10])
//      OPTRK = linevw.(thefields[12])
//      PKCAP = linevw.(thefields[13])
//      BPRA = linevw.(thefields[14])
//      BPRB = linevw.(thefields[15])
//      FFTMIN = linevw.(thefields[16])
//      MDT = linevw.(thefields[17])
//      MDAT = linevw.(thefields[18])
//      MDTT = linevw.(thefields[19])
//      AT = linevw.(thefields[21])
//      US41ID = linevw.(thefields[22])
//      BridgeID = linevw.(thefields[23])
//      COR = linevw.(thefields[24])
//      CO = linevw.(thefields[26])
//      RTENM = linevw.(thefields[27])
//      CO = Left(RTENM, 2)
//      FTFC = linevw.(thefields[28])
//      ABPKCAP = linevw.(thefields[29])
//      BAPKCAP = linevw.(thefields[30])

if ABPKCAP = null then ABPKCAP = 0
if BAPKCAP = null then BAPKCAP = 0
PKCAP = 2 * Max(ABPKCAP, BAPKCAP) / 3

fhwa_fc = FTFC
totln = 2 * linevw.LN1DIR
if ABPKCAP = null | ABPKCAP = 0 | BAPKCAP = null | BAPKCAP = 0 then totln = linevw.LN1DIR
ln = linevw.LN
if (ln = 5 or ln = 7 or ln = 9) then divided = true
ln_one_dir = linevw.LN1DIR

// load HCM ideal capacities
if fhwa_fc = 12 then idealcap = 2000 // freeways and expressways
if (fhwa_fc = 1 or fhwa_fc = 11) then do
    if ln_one_dir <= 2 then idealcap = 2200 // interstates
    if ln_one_dir > 2 then idealcap = 2300
end
if ((fhwa_fc = 2 or fhwa_fc = 6) and totln > 2 and divided) then do
    idealcap = 1600 // arterials/multilane divided
end

```

```

if ((fhwa_fc = 14 or fhwa_fc = 16) and totln > 2 and divided) then do
    idealcap = 1200 // arterials/multilane divided
end
if ((fhwa_fc = 2 or fhwa_fc = 6) and totln > 2 and not divided) then do
    idealcap = 1400 // arterials/multilane undivided
end
if ((fhwa_fc = 14 or fhwa_fc = 16) and totln > 2 and not divided) then do
    idealcap = 1200 // arterials/multilane undivided
end
if ((fhwa_fc = 7 or fhwa_fc = 8 or fhwa_fc = 9) and totln > 2 and divided) then do
    idealcap = 1400 // collectors and local/multilane divided
end
if ((fhwa_fc = 17 or fhwa_fc = 19) and totln > 2 and divided) then do
    idealcap = 1200 // collectors and local/multilane divided
end
if ((fhwa_fc = 7 or fhwa_fc = 8 or fhwa_fc = 9) and totln > 2 and not divided) then do
    idealcap = 1400 // collectors and multilane undivided
end
if ((fhwa_fc = 17 or fhwa_fc = 19) and totln > 2 and not divided) then do
    idealcap = 1200 // collectors and multilane undivided
end
if ((fhwa_fc = 2 or fhwa_fc = 6) and totln = 2) then do
    idealcap = 1400 // 2 lane arterials
end
if ((fhwa_fc = 14 or fhwa_fc = 16) and totln = 2) then do
    idealcap = 1200 // 2 lane arterials
end
if ((fhwa_fc = 7 or fhwa_fc = 8 or fhwa_fc = 9 or fhwa_fc = 17 or fhwa_fc = 19) and totln = 2) then do
    idealcap = 1200 // 2 lane collectors and local
end
if ((fhwa_fc = 7 or fhwa_fc = 8 or fhwa_fc = 9 or fhwa_fc = 17 or fhwa_fc = 19) and totln = 1) then do
    idealcap = 1200 // 2 lane collectors and local
end

PKCAP = idealcap * LANES

If np > 3 then do
    MDAUTO = linevw.(thefields[7])
    MDTRK = linevw.(thefields[11])
end
else do
    MDAUTO = 0
    MDTRK = 0
end

// Exclude centroid connectors
if FC = 99 then goto Skip

//Avoid adding nulls problem

IF AMAUTO = null then AMAUTO = 0
IF PMAUTO = null then PMAUTO = 0
IF MDAUTO = null then MDAUTO = 0
IF OPAUTO = null then OPAUTO = 0
IF AMTRK = null then AMTRK = 0
IF PMTRK = null then PMTRK = 0
IF MDTRK = null then MDTRK = 0
IF OPTRK = null then OPTRK = 0

//Reset daily period volume arrays to zero for each period

for i = 1 to dp do
    PCEVOL[i] = 0
    ADTVOL[i] = 0
    CARVOL[i] = 0
    TRKVOL[i] = 0
end

//Before beginning daily loop
if todass = 1 then do
    PCEAM = (AMTRK*1.5) + AMAUTO
    PCEPM = (PMTRK*1.5) + PMAUTO
    PCEMD = (MDTRK*1.5) + MDAUTO
    PCEOP = (OPTRK*1.5) + OPAUTO
    PCES = {PCES, PCEAM, PCEPM, PCEMD}

    ADTAM = AMTRK + AMAUTO
    ADTPM = PMTRK + PMAUTO
    ADTMD = MDTRK + MDAUTO
    ADTOP = OPTRK + OPAUTO
    ADTS = {ADTOP, ADTAM, ADTPM, ADTMD}

    CARS = {OPAUTO, AMAUTO, PMAUTO, MDAUTO}
    TRKS = {OPTRK, AMTRK, PMTRK, MDTRK}
end
dim PCE[dp]
dim ADT[dp]
dim CAR[dp]
dim TRK[dp]

if todass = 1 then do
    for i = 1 to dp do
        tp = 1 + PERIODS[i]
        PCE[i] = PCES[tp]
        ADT[i] = ADTS[tp]
        CAR[i] = CARS[tp]
        TRK[i] = TRKS[tp]
    end
end

if todass = 0 then do

```

```

        for i = 1 to dp do
            PCE[i] = (MDTT*1.5) + MDAT
            ADT[i] = MDT
            CAR[i] = MDAT
            TRK[i] = MDTT
        end
    end

    DPCAP = PKCAP * 0.5 * (24/dp)

    alpha = 0.15
    beta = 4.0
    if BPRA <> null then alpha = BPRA
    if BPRB <> null then beta = BPRB

    delay = 0
    AMTIME = 0
    PMTIME = 0
    MDTIME = 0
    OPTIME = 0
    AMADTVOL = 0
    PMADTVOL = 0
    MDADTVOL = 0
    OPADTVOL = 0
    AMSP = 0
    PMSP = 0
    MDSP = 0
    OPSP = 0
    AMMAXVC = 0
    PMMAXVC = 0
    MDMAXVC = 0
    OPMAXVC = 0
    FX = 0
    IX = 0
    PX = 0
    TX = 0

//Initialize arrays

    dim VC[dp]
    dim time[dp]
    dim losttime[dp]
    dim sp[dp]
    dim VTHR[dp]
    dim VTHR[dp]
    dim ATHR[dp]
    dim AMTHR[dp]
    dim TTHR[dp]
    dim TMTHR[dp]

//Begin daily loop to calculate hourly volumes, v/c ratios, and
//congested travel times and speeds.

    for i = 1 to dp do

        // Compute hourly volumes

        PCEVOL[i] = PCE[i] * DDIST[i]
        ADTVOL[i] = ADT[i] * DDIST[i]
        CARVOL[i] = CAR[i] * DDIST[i]
        TRKVOL[i] = TRK[i] * DDIST[i]

        // Compute hourly v/c ratio

        VC[i] = PCEVOL[i] * PDIR[i] / DPCAP

        // Compute hourly travel times and delays

        time[i] = FFTMIN * (1 + (alpha * pow(VC[i], beta)))

        losttime[i] = time[i] - FFTMIN

        delay = delay + losttime[i] * ADTVOL[i]

        // Begin calculation of average travel times and speeds for the periods

        sp[i] = (LENGTH / time[i]) * 60
        if sp[i] < 0.01 then sp[i] = 0.01

        if PERIODS[i] = 1 then do
            AMTIME = AMTIME + (time[i] * ADTVOL[i])
            AMADTVOL = AMADTVOL + ADTVOL[i]
            AMSP = AMSP + (sp[i] * ADTVOL[i])
            AMMAXVC = Max(AMMAXVC, VC[i])
        end

        if PERIODS[i] = 2 then do
            PMTIME = PMTIME + (time[i] * ADTVOL[i])
            PMADTVOL = PMADTVOL + ADTVOL[i]
            PMSP = PMSP + (sp[i] * ADTVOL[i])
            PMMAXVC = Max(PMMAXVC, VC[i])
        end

        if PERIODS[i] = 3 then do
            MDTIME = MDTIME + (time[i] * ADTVOL[i])
            MDADTVOL = MDADTVOL + ADTVOL[i]
            MDSP = MDSP + (sp[i] * ADTVOL[i])
            MDMAXVC = Max(MDMAXVC, VC[i])
        end

        if PERIODS[i] = 0 then do

```

```

        OPTIME = OPTIME + (time[i] * ADTVOL[i])
        OPADTVOL = OPADTVOL + ADTVOL[i]
        OPSP = OPSP + (sp[i] * ADTVOL[i])
        OPMAXVC = Max(OPMAXVC, VC[i])
    end

    // Compute hourly VMT & VHT

    VHTHR[i] = (ADTVOL[i] * time[i]) / 60
    VMTHR[i] = ADTVOL[i] * LENGTH
    AHTHR[i] = (CARVOL[i] * time[i]) / 60
    AMTHR[i] = CARVOL[i] * LENGTH
    THTHR[i] = (TRKVOL[i] * time[i]) / 60
    TMTTHR[i] = TRKVOL[i] * LENGTH

    // Compute congested VMT & VHT

    For j = 1 to Ls do
        If VC[i] > CLEVELS[j] then do
            VMTVC[j] = VMTVC[j] + VMTHR[i]
            VHTVC[j] = VHTVC[j] + VHTHR[i]
            AMTVC[j] = AMTVC[j] + AMTHR[i]
            AHTVC[j] = AHTVC[j] + AHTHR[i]
            TMTVC[j] = TMTVC[j] + TMTTHR[i]
            THTVC[j] = THTVC[j] + THTHR[i]
        end
    end
//      end daily loop

//After daily loop
//Calculate total system delay

    If delay <> null then Tot_Delay = Tot_Delay + (delay/60)

//Complete calculation of average travel times and speeds for the periods and the day
//Remember you must avoid division by zero

    If AMADTVOL < 1 then AMADTVOL = 1
    If PMADTVOL < 1 then PMADTVOL = 1
    If MDADTVOL < 1 then MDADTVOL = 1
    If OPADTVOL < 1 then OPADTVOL = 1

    AMAVGTT = AMTIME / AMADTVOL
    AMAVGSP = AMSP / AMADTVOL
    PMAVGTT = PMTIME / PMADTVOL
    PMAVGSP = PMSP / PMADTVOL
    MDAVGTT = MDTIME / MDADTVOL
    MDAVGSP = MDSP / MDADTVOL
    OPAVGTT = OPTIME / OPADTVOL
    OPAVGSP = OPSP / OPADTVOL
    AVGTT = (AMTIME + PMTIME + MDTIME + OPTIME) /
        (AMADTVOL + PMADTVOL + MDADTVOL + OPADTVOL)

    BAD = 0
    If AVGTT < FFTMIN then do
        BAD = 1
        AVGTT = FFTMIN
    end

    AVGSP = (LENGTH / AVGTT)* 60

// What is maximum v/c ratio and the worst LOS experienced on the link?

    MAXVC = Max(OPMAXVC, Max(MDMAXVC, Max(AMMAXVC, PMMAXVC)))
    MAXMAXVC = Max(MAXMAXVC, MAXVC)

    If FC = null then FC = FTFC
    if FC = 1 or FC = 11 then do
        LOS = "A"
        IF MAXVC > .29 THEN LOS = "B"
        IF MAXVC > .47 THEN LOS = "C"
        IF MAXVC > .69 THEN LOS = "D"
        IF MAXVC > .88 THEN LOS = "E"
        IF MAXVC > 1.00 THEN LOS = "F"
    end
    if FC = 12 then do
        LOS = "A"
        IF MAXVC > .33 THEN LOS = "B"
        IF MAXVC > .55 THEN LOS = "C"
        IF MAXVC > .75 THEN LOS = "D"
        IF MAXVC > .88 THEN LOS = "E"
        IF MAXVC > 1.00 THEN LOS = "F"
    end
    if FC = 2 or FC = 6 or FC = 14 or FC = 16 then do
        LOS = "A"
        IF MAXVC > .30 THEN LOS = "B"
        IF MAXVC > .50 THEN LOS = "C"
        IF MAXVC > .70 THEN LOS = "D"
        IF MAXVC > .84 THEN LOS = "E"
        IF MAXVC > 1.00 THEN LOS = "F"
    end
    if FC = 7 or FC = 8 or FC = 9 or FC = 17 or FC = 19 then do
        LOS = "A"
        IF MAXVC > .31 THEN LOS = "B"
        IF MAXVC > .52 THEN LOS = "C"
        IF MAXVC > .72 THEN LOS = "D"
        IF MAXVC > .83 THEN LOS = "E"
        IF MAXVC > 1.00 THEN LOS = "F"
    end

```

```

//Calculate total road miles, total road lane miles, and congested road lane miles

    RM = RM + LENGTH
    RLM = RLM + (LENGTH * LANES)

    For j = 1 to Ls do
        If MAXVC > CLEVELS[j] then do
            CLANEMI[j] = CLANEMI[j] + (LENGTH * LANES)
        end
    end

// Calculate VHT and VMT by mode

    VHT = (MDT * AVGTT)/60
    VHT_Trk = (MDTT * AVGTT)/60
    VHT_Auto = (MDAT * AVGTT)/60

    VMT = MDT * LENGTH
    VMT_Trk = MDTT * LENGTH
    VMT_Auto = MDAT * LENGTH

    IF VHT <> null then TOTVHT = TOTVHT + VHT
    IF VHT_Trk <> null then TOTVHT_Trk = TOTVHT_Trk + VHT_Trk
    IF VHT_Auto <> null then TOTVHT_Auto = TOTVHT_Auto + VHT_Auto

    IF VMT <> null then TOTVMT = TOTVMT + VMT
    IF VMT_Trk <> null then TOTVMT_Trk = TOTVMT_Trk + VMT_Trk
    IF VMT_Auto <> null then TOTVMT_Auto = TOTVMT_Auto + VMT_Auto

// Calculate Accidents by type

    If FC = null then FC = FTFC
    FX = 330*VMT*FXR[FC]/100000000
    IX = 330*VMT*IXR[FC]/100000000
    PX = 330*VMT*PXR[FC]/100000000
    TX = 330*VMT*TXR[FC]/100000000

    If FX <> null then TOTFX = TOTFX + FX
    If IX <> null then TOTIX = TOTIX + IX
    If PX <> null then TOTPX = TOTPX + PX
    If TX <> null then TOTTX = TOTTX + TX

//Begin Functional Class Loop
// Remember i is not equal to the functional class number

    for i = 1 to FCLASS.length do

        If FC = FCLASS[i] then do
            WFCVC = MAXVC * VMT
            CVC[i] = CVC[i] + WFCVC
            TOTVC = TOTVC + WFCVC
            RDMILES[i] = RDMILES[i] + LENGTH
            VMTC[i] = VMTC[i] + VMT
            VHTC[i] = VHTC[i] + VHT
            VMTAUTO[i] = VMTAUTO[i] + VMT_Auto
            VHTAUTO[i] = VHTAUTO[i] + VHT_Auto
            VMTRK[i] = VMTRK[i] + VMT_Trk
            VHTRK[i] = VHTRK[i] + VHT_Trk
            SUMFX[i] = SUMFX[i] + FX
            SUMIX[i] = SUMIX[i] + IX
            SUMPX[i] = SUMPX[i] + PX
            SUMTX[i] = SUMTX[i] + TX
        end
    end
//      end Functional Class Loop

//Begin Area Type Loop

    for i = 1 to AREA.length do

        j = i + 16
        If AT = AREA[i] then do
            WACVC = MAXVC * VMT
            CVC[j] = CVC[j] + WACVC
            RDMILES[j] = RDMILES[j] + LENGTH
            VMTC[j] = VMTC[j] + VMT
            VHTC[j] = VHTC[j] + VHT
            VMTAUTO[j] = VMTAUTO[j] + VMT_Auto
            VHTAUTO[j] = VHTAUTO[j] + VHT_Auto
            VMTRK[j] = VMTRK[j] + VMT_Trk
            VHTRK[j] = VHTRK[j] + VHT_Trk
            SUMFX[j] = SUMFX[j] + FX
            SUMIX[j] = SUMIX[j] + IX
            SUMPX[j] = SUMPX[j] + PX
            SUMTX[j] = SUMTX[j] + TX
        end
    end
//      end Area Type Loop

//Begin County Loop

    for i = 1 to COS.length do

        j = i + fc2 - 1
        If CO = COS[i] then do
            WCOCVC = MAXVC * VMT
            CVC[j] = CVC[j] + WCOCVC
            RDMILES[j] = RDMILES[j] + LENGTH
            VMTC[j] = VMTC[j] + VMT
            VHTC[j] = VHTC[j] + VHT

```

```

        VMTAUTO[j] = VMTAUTO[j] + VMT_Auto
        VHTAUTO[j] = VHTAUTO[j] + VHT_Auto
        VMTTRK[j] = VMTTRK[j] + VMT_Trk
        VHTTRK[j] = VHTTRK[j] + VHT_Trk
        SUMFX[j] = SUMFX[j] + FX
        SUMIX[j] = SUMIX[j] + IX
        SUMPX[j] = SUMPX[j] + PX
        SUMTX[j] = SUMTX[j] + TX
    end
end
//      end County Loop

//Begin Corridor Loop

for i = 1 to CORRS.length do

    j = i + fc3 - 1
    If COR = CORRS[i] then do
        WCCVC = MAXVC * VMT
        CVC[j] = CVC[j] + WCCVC
        RDMILES[j] = RDMILES[j] + LENGTH
        VMTC[j] = VMTC[j] + VMT
        VHTC[j] = VHTC[j] + VHT
        VMTAUTO[j] = VMTAUTO[j] + VMT_Auto
        VHTAUTO[j] = VHTAUTO[j] + VHT_Auto
        VMTTRK[j] = VMTTRK[j] + VMT_Trk
        VHTTRK[j] = VHTTRK[j] + VHT_Trk
        SUMFX[j] = SUMFX[j] + FX
        SUMIX[j] = SUMIX[j] + IX
        SUMPX[j] = SUMPX[j] + PX
        SUMTX[j] = SUMTX[j] + TX
    end
end
//      end Corridor Loop

//VMT and VHT Along US-41 (I-64 to Pennyrile at KY 425)

if US41ID = 1 then do
    VHT_US41 = VHT_US41 + VHT
    VHTTRK_US41 = VHTTRK_US41 + VHT_Trk
    VHTAUTO_US41 = VHTAUTO_US41 + VHT_Auto
    VMT_US41 = VMT_US41 + VMT
    VMTTRK_US41 = VMTTRK_US41 + VMT_Trk
    VMTAUTO_US41 = VMTAUTO_US41 + VMT_Auto
end

//VPD (VEHICLES PER DAY ON US41/OHIO-RIVER BRIDGE)AADT on the Bridge

if BridgeID = 1 then do
    VPDAUTO_Bridge = Round(MDAT,0)
    VPDTRK_Bridge = Round(MDTT,0)
    VPDTOT_Bridge = Round(MDT,0)
end

//Write link information into the output link file

AddRecord(linkfile,
          {{"LID", "ID"}, {"FHWA_FC", "FC"}, {"MAXVC1", "MAXVC"}, {"AVGSP", "AVGSP"}, {"AVGTT1", "AVGTT"}, {"WORSTLOS", "LOS"}, {"HRS_DELAY", "delay"}, {"VHT", "VHT"}, {"VMT", "VMT"}, {"BPRA", "alpha"}, {"BPRE", "beta"}, {"FFTMIN", "FFTMIN"}, {"BAD", "BAD"}, {"AMAVGTT", "AMAVGTT"}, {"PMAVGTT", "PMAVGTT"}, {"OPAVGTT", "OPAVGTT"}, {"FX", "FX"}, {"IX", "IX"}, {"PX", "PX"}, {"TX", "TX"}))

//Write link information into the output Mobile6 file

dim marec[73]
marec[1] = {"LID", "ID"}
for i = 1 to 24 do
    a = i + 1
    b = i + 25
    c = i + 49
    marec[a] = {carvolfield[i], CARVOL[i]}
    marec[b] = {trkvoldfield[i], TRKVOL[i]}
    marec[c] = {avgspdfield[i], sp[i]}
end

AddRecord(mfile, marec)

Skip:
rec = GetNextRecord(linevw + "|", null, null)
end
//end loop through assigned network

UpdateProgressBar("Writing Report", 99)

//Calculation of the ESPI Indices

```

```

If MAXMAXVVC > 0.70 then do
    ESPI_VMT = (10*TOTVMT) / (VMTVC[2] + VMTVC[4])
    ESPI_VHT = (10*TOTVHT) / (VHTVC[2] + VHTVC[4])
    ESPI_VMT_CAR = (10*TOTVMT_Auto) / (AMTVC[2] + AMTVC[4])
    ESPI_VHT_CAR = (10*TOTVHT_Auto) / (AHTVC[2] + AHTVC[4])
    ESPI_VMT_TRK = (10*TOTVMT_Trk) / (TMTVC[2] + TMTVC[4])
    ESPI_VHT_TRK = (10*TOTVHT_Trk) / (THTVC[2] + THTVC[4])
end

//Normalize the Functional Class, Area Type, Corridor, and System v/c Ratios
//And Calculate Avg Speeds by Functional Class, Area Type, and Corridor

for i = 1 to FCLASS.length do
    if VMTC[i] <> 0 then CVC[i] = Round(CVC[i]/VMTC[i], 2)
    if VHTC[i] <> 0 then SPDC[i] = Round(VMTC[i] / VHTC[i], 2)
    if VHTC[i] = 0 then SPDC[i] = 0.00
end
for i = 1 to AREA.length do
    j = i + 16
    if VMTC[j] <> 0 then CVC[j] = Round(CVC[j]/VMTC[j], 2)
    if VHTC[j] <> 0 then SPDC[j] = Round(VMTC[j] / VHTC[j], 2)
    if VHTC[j] = 0 then SPDC[j] = 0.00
end
for i = 1 to COS.length do
    j = i + fc2 - 1
    if VMTC[j] <> 0 then CVC[j] = Round(CVC[j]/VMTC[j], 2)
    if VHTC[j] <> 0 then SPDC[j] = Round(VMTC[j] / VHTC[j], 2)
    if VHTC[j] = 0 then SPDC[j] = 0.00
end
for i = 1 to CORRS.length do
    j = i + fc3 - 1
    if VMTC[j] <> 0 then CVC[j] = Round(CVC[j]/VMTC[j], 2)
    if VHTC[j] <> 0 then SPDC[j] = Round(VMTC[j] / VHTC[j], 2)
    if VHTC[j] = 0 then SPDC[j] = 0.00
end
TOTVC = TOTVC / TOTVMT

//Write Summary File

dim systat[Ns]
systat[1] = "Total VMT"
systat[2] = "VMT with v/c > 0.50"
systat[3] = "VMT with v/c > 0.70"
systat[4] = "VMT with v/c > 0.84"
systat[5] = "VMT with v/c > 0.99"
systat[6] = "Total VHT"
systat[7] = "VHT with v/c > 0.50"
systat[8] = "VHT with v/c > 0.70"
systat[9] = "VHT with v/c > 0.84"
systat[10] = "VHT with v/c > 0.99"

systat[12] = "Total Road Miles"
systat[13] = "Total Lane Miles"
systat[14] = "Lane Miles with v/c > 0.50"
systat[15] = "Lane Miles with v/c > 0.70"
systat[16] = "Lane Miles with v/c > 0.84"
systat[17] = "Lane Miles with v/c > 0.99"

systat[19] = "Total VMT on US 41"
systat[20] = "Auto VMT on US 41"
systat[21] = "Truck VMT on US 41"
systat[22] = "Total VHT on US 41"
systat[23] = "Auto VHT on US 41"
systat[24] = "Truck VHT on US 41"
systat[25] = "Total VPD on Twin Bridges"
systat[26] = "Auto VPD on Twin Bridges"
systat[27] = "Truck VPD on Twin Bridges"

systat[29] = "ESPI by VMT"
systat[30] = "ESPI by VHT"
systat[31] = "ESPI by Auto VMT"
systat[32] = "ESPI by Auto VHT"
systat[33] = "ESPI by Truck VMT"
systat[34] = "ESPI by Truck VHT"

systat[36] = "Total Systemwide Delay"

dim cstat[Ns]
cstat[1] = i2s(Round(TOTVMT, 0))
cstat[2] = i2s(Round(VMTVC[1], 0))
cstat[3] = i2s(Round(VMTVC[2], 0))
cstat[4] = i2s(Round(VMTVC[3], 0))
cstat[5] = i2s(Round(VMTVC[4], 0))
cstat[6] = i2s(Round(TOTVHT, 0))
cstat[7] = i2s(Round(VHTVC[1], 0))
cstat[8] = i2s(Round(VHTVC[2], 0))
cstat[9] = i2s(Round(VHTVC[3], 0))
cstat[10] = i2s(Round(VHTVC[4], 0))

cstat[12] = i2s(Round(RM, 0))
cstat[13] = i2s(Round(RLM, 0))
cstat[14] = i2s(Round(CLANEMI[1], 0))
cstat[15] = i2s(Round(CLANEMI[2], 0))
cstat[16] = i2s(Round(CLANEMI[3], 0))
cstat[17] = i2s(Round(CLANEMI[4], 0))

cstat[19] = i2s(Round(VMT_US41, 0))
cstat[20] = i2s(Round(VMTAUTO_US41, 0))
cstat[21] = i2s(Round(VMTTRK_US41, 0))
cstat[22] = i2s(Round(VHT_US41, 0))
cstat[23] = i2s(Round(VHTAUTO_US41, 0))

```

```

cstat[24] = i2s(Round(VHTTRK_US41, 0))
cstat[25] = i2s(Round(VPDTOT_Bridge, 0))
cstat[26] = i2s(Round(VPDAUTO_Bridge, 0))
cstat[27] = i2s(Round(VPDTRK_Bridge, 0))

cstat[29] = r2s(Round(ESPI_VMT, 2))
cstat[30] = r2s(Round(ESPI_VHT, 2))
cstat[31] = r2s(Round(ESPI_VMT_CAR, 2))
cstat[32] = r2s(Round(ESPI_VHT_CAR, 2))
cstat[33] = r2s(Round(ESPI_VMT_TRK, 2))
cstat[34] = r2s(Round(ESPI_VHT_TRK, 2))

cstat[36] = i2s(Round(Tot_Delay, 0))

dim class[Ns]
class[1] = "Rural Interstates"
class[2] = "Other Rural Principal Arterials"
class[3] = "Rural Minor Arterials"
class[4] = "Rural Major Collectors"
class[5] = "Rural Minor Collectors"
class[6] = "Rural Local Roads"
class[7] = "Urban Interstates"
class[8] = "Other Urban Freeways & Expwys"
class[9] = "Other Urban Principal Arterials"
class[10] = "Urban Minor Arterials"
class[11] = "Urban Collectors"
class[12] = "Urban Local Roads"
class[14] = "-----Totals-----"
class[17] = "Area Type --- CBD"
class[18] = "Area Type --- URB"
class[19] = "Area Type --- SUB"
class[20] = "Area Type --- RUR"
class[23] = "County --- Gibson"
class[24] = "County --- Henderson"
class[25] = "County --- Posey"
class[26] = "County --- Vanderburgh"
class[27] = "County --- Warrick"
class[30] = "Corridor --- Lloyd Expwy (in Vand. Co)"
class[31] = "Corridor --- US 41 (IN only)"
class[32] = "Corridor --- US 41 Bridge"
class[33] = "Corridor --- I-164"
class[34] = "Corridor --- I-64"
class[35] = "Corridor --- SR-57"
class[36] = "Corridor --- Pennyrike Pkwy"

datetime = GetDateAndTime()
class[40] = "Network Layer: " + linevw
class[41] = "Network File: " + netfilename
class[42] = "Report File: " + outfilename
class[43] = datetime

FFC = {1,2,6,7,8,9,11,12,14,16,17,19,.....}

RDMILES[14] = Round(RM, 0)
VMTC[14] = Round(TOTVMT, 0)
VMTAUTO[14] = Round(TOTVMT_Auto, 0)
VMTRRK[14] = Round(TOTVMT_Trk, 0)
VHTC[14] = Round(TOTVHT, 0)
VHTAUTO[14] = Round(TOTVHT_Auto, 0)
VHTTRK[14] = Round(TOTVHT_Trk, 0)
SPDC[14] = Round(TOTVMT/TOTVHT, 2)
CVC[14] = Round(TOTVC, 2)
SUMFX[14] = TOTFX
SUMIX[14] = TOTIX
SUMPX[14] = TOTPX
SUMTX[14] = TOTTX

for i = 1 to Ns do

    AddRecord(postrep,      {"SystemStat", systat[i]},
              {"Stat", cstat[i]},
              {"Class", class[i]},
              {"FHWA_FC", FFC[i]},
              {"RoadMiles", Round(RDMILES[i],2)},
              {"VMT", Round(VMTC[i],0)},
              {"Auto_VMT", Round(VMTAUTO[i],0)},
              {"Truck_VMT", Round(VMTRRK[i],0)},
              {"VHT", Round(VHTC[i],0)},
              {"Auto_VHT", Round(VHTAUTO[i],0)},
              {"Truck_VHT", Round(VHTTRK[i],0)},
              {"AvgSpeed", Round(SPDC[i],2)},
              {"Vc", Round(CVC[i],2)},
              {"Fatal_Xs", Round(SUMFX[i],0)},
              {"Injury_Xs", Round(SUMIX[i],0)},
              {"PDO_Xs", Round(SUMPX[i],0)},
              {"Total_Xs", Round(SUMTX[i],0)})}

end

CloseView(linkfile)
CloseView(mfile)
CloseView(postrep)

DestroyProgressBar()
SetStatus(i, "@System0", )
quit:
endMacro

macro "addavgtt" (in_value)
  thepath = in_value[1]

```

```

linevw = in_value[2]
postgar = LoadArray(theopath + "\\post\\post.gar")
linkfilename = postgar[1]
totrec = GetRecordCount(linevw)

thefields1 = {"AVGTT", "AVGSPD", "AVGTTAM", "AVGTPM", "AVGTOFF"}
RunMacro("addfields", (thefields1, linevw))

EnableProgressBar("Post_Alt", 1)
CreateProgressBar("loading...", "False")
recno = 0

paview = OpenTable("postlinks", "dBASE", (linkfilename, ))
joinvw = JoinViews("Network+Post_Alt", linevw + ".ID", paview + ".LID", )
jnrec = GetFirstRecord(joinvw + "|", null)
while jnrec <> null do

    recno = recno + 1
    prog = Round ((recno/totrec)*99, 0)
    UpdateProgressBar("Updating Network...", prog)

    joinvw.AVGTT = joinvw.AVGTT1
    joinvw.AVGSPD = joinvw.AVGSP
    joinvw.AVGTAM = joinvw.AMAVGT
    joinvw.AVGTPM = joinvw.PMAVGT
    joinvw.AVGTOFF = joinvw.OPAVGTT
    jnrec = GetNextRecord(joinvw + "|", null, null)
end

CloseView(joinvw)
CloseView(paview)

DestroyProgressBar()

endmacro

macro "addlos" (in_value)
theopath = in_value[1]
linevw = in_value[2]

postgar = LoadArray(theopath + "\\post\\post.gar")
linkfilename = postgar[1]

totrec = GetRecordCount(linevw)

fldnames2 = {"MAXVC", "LOS"}
fldtype = {"Real", 10, 2}, {"String", 3, null}

struct = GetTableStructure(linevw)
viewflds = GetFields(linevw, numeric)

for i = 1 to struct.length do
    struct[i] = struct[i] + {struct[i][1]}
end

for i = 1 to fldnames2.length do
    pos = ArrayPosition(viewflds[1], {fldnames2[i]}, )
    if pos = 0 then do
        newstr = newstr +
            {(fldnames2[i], fldtype[i][1], fldtype[i][2], fldtype[i][3], "false", null, null, null, null)}
        modtab = 1
    end
end

if modtab = 1 then do
    newstr = struct + newstr
    ModifyTable(in_value[2], newstr)
end

EnableProgressBar("Post_Alt", 1)
CreateProgressBar("loading...", "False")
recno = 0

paview = OpenTable("postlinks", "dBASE", (linkfilename, ))
joinvw = JoinViews("Network+Post_Alt", linevw + ".ID", paview + ".LID", )
jnrec = GetFirstRecord(joinvw + "|", null)
while jnrec <> null do

    recno = recno + 1
    prog = Round ((recno/totrec)*99, 0)
    UpdateProgressBar("Updating Network...", prog)

    joinvw.MAXVC = joinvw.MAXVC1
    joinvw.LOS = joinvw.WORSTLOS
    jnrec = GetNextRecord(joinvw + "|", null, null)
end

CloseView(joinvw)
CloseView(paview)

DestroyProgressBar()

endmacro

Macro "4M6" (in_value)
theopath = in_value[1]
linevw = in_value[2]

```

```

year = in_value[3]

/*
A Program written by Vincent L. Bernardin, Jr. of Bernardin, Lochmueller, and Assoc. to post-process the EUTS regional model to
calculate the VMT by facility type, by hour, and by speed bin for input into the Mobile6 air quality model using the *24M6.dbf
file created by Post_Alt.
6/11/03
Revised 6/23/03
*/
postgar = LoadArray(theopath + "\\post\\post.gar")
mfile = postgar[2]
retterr = 0

//Set the analysis year
yn = 0
if year = 1 then yn = 1
if year = 2 then yn = 2
if year = 3 then yn = 3
if year > 3 then yn = 4
if yn = 0 then do
    ShowMessage("Error: Invalid Analysis Year!")
    retterr = retterr + 1
    goto quit2
end

//Get Line Layer and record count
//
//      linevw = GetLayer()
//      ShowMessage("The current layer is " + linevw)
//      totrec = GetRecordCount(linevw,)

//Choose output directory
//      theopath = ChooseDirectory("Choose a directory for the results")

//Read in the Network Field Names from the auxillary file
//      thefields = RunMacro("getnames", {theopath + "\\post\\\"})

//Choose and open the input file from Post_Alt
//      mfile = ChooseFile({{"24M6 File", "*_24M6.dbf"}, 
//                          "Choose the input file",
//                          {"Initial Directory", theopath},})
//      mvw = OpenTable("24M6view", "DBASE", {mfile, })

EnableProgressBar("4M6", 1)
CreateProgressBar("loading...", "False")

//Define Mobile6 speed bins
M6spdbin = {2.5, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65}

//Initialize Arrays
dim CLASSES[4]
dim C_VMT[24]
dim T_VMT[24]
dim FVMT1[2]
dim HVMT1[3]
dim SVMT1[2]
dim FVMT3[28]
dim HVMT2[3]
dim NAR24[24]
dim NAR5[5]
dim NAR15[15]
dim NAR28[28]
dim CARVOL[24]
dim TRKVOL[24]
dim AVGSPD[24]
dim HBVMT[2]

//Zero out arrays to be summed
for m = 1 to 2 do
    FVMT1[m] = CopyArray(NAR24)
    for h = 1 to 24 do
        FVMT1[m][h] = CopyArray(NAR5)
        for f = 1 to 5 do
            FVMT1[m][h][f] = 0
        end
    end
end
for m = 1 to 28 do
    FVMT3[m] = CopyArray(NAR24)
    for h = 1 to 24 do
        FVMT3[m][h] = CopyArray(NAR5)
    end
end
for q = 1 to 3 do
    HVMT2[q] = CopyArray(NAR24)
    HVMT1[q] = CopyArray(NAR24)
    for i = 1 to 24 do
        HVMT1[q][i] = 0
        HVMT2[q][i] = 0
    end
end
for f = 1 to 2 do
    SVMT1[f] = CopyArray(NAR24)
    for h = 1 to 24 do
        SVMT1[f][h] = CopyArray(NAR15)
    end
end

```

```

        for s = 1 to 15 do
            SVMT1[f][h][s] = 0
        end
    end
    for f = 1 to 2 do
        HBVMT[f] = CopyArray(NAR15)
        for s = 1 to 14 do
            HBVMT[f][s] = CopyArray(NAR24)
        end
    end
//Define the default mode to class breakout

MODE = {1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 1,
        1, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 1}

CLASSES[1] = {0.680, 0.048, 0.162, 0.064, 0.030, 0.253, 0.010, 0.010,
               0.020, 0.040, 0.030, 0.000, 0.000, 0.006, 0.002, 0.071,
               0.020, 0.010, 0.001, 0.030, 0.051, 0.091, 0.333, 0.007,
               0.010, 0.010, 0.010, 0.001}

CLASSES[2] = {0.667, 0.050, 0.170, 0.066, 0.030, 0.252, 0.009, 0.019,
               0.019, 0.047, 0.028, 0.001, 0.000, 0.006, 0.002, 0.065,
               0.019, 0.009, 0.001, 0.028, 0.056, 0.093, 0.327, 0.007,
               0.009, 0.009, 0.009, 0.002}

CLASSES[3] = {0.654, 0.053, 0.178, 0.068, 0.032, 0.250, 0.017, 0.017,
               0.017, 0.043, 0.034, 0.000, 0.000, 0.005, 0.003, 0.069,
               0.017, 0.009, 0.000, 0.026, 0.052, 0.095, 0.327, 0.006,
               0.009, 0.009, 0.009, 0.001}

CLASSES[4] = {0.649, 0.054, 0.183, 0.068, 0.032, 0.254, 0.017, 0.017,
               0.017, 0.042, 0.034, 0.000, 0.000, 0.005, 0.003, 0.068,
               0.017, 0.008, 0.001, 0.025, 0.051, 0.093, 0.332, 0.005,
               0.008, 0.008, 0.008, 0.001}

//Create the 24M6field arrays
dim carvolfield[24]
dim trkvolfield[24]
dim avgspdfield[24]
for i = 1 to 24 do
    if i > 9 then do
        carvolfield[i] = "CARVOL_" + i2s(i)
        trkvolfield[i] = "TRKVOL_" + i2s(i)
        avgspdfield[i] = "AVGSPD_" + i2s(i)
    end
    else do
        carvolfield[i] = "CARVOL_0" + i2s(i)
        trkvolfield[i] = "TRKVOL_0" + i2s(i)
        avgspdfield[i] = "AVGSPD_0" + i2s(i)
    end
end

//Create, Open, and begin Looping the Joined View

dview = JoinViews("Network+24M6", linevw + ".ID", mvw + ".LID", )

SetView(dview)
recno = 0
rec = GetFirstRecord(dview + "|", null)
while rec <> null do

    recno = recno + 1
    prog = Round ((recno/totrec)*99, 0)
    UpdateProgressBar("Computing Statistics", prog)
    SetStatus(1, "Record " + i2s(recno) + " of " + i2s(totrec), )

//Read in 24M6 data arrays

    for i = 1 to 24 do
        CARVOL[i] = dview.(carvolfield[i])
        TRKVOL[i] = dview.(trkvolfield[i])
        AVGSPD[i] = dview.(avgspdfield[i])
    end
    if recno = 1 then do
        ShowArray(CARVOL)
        ShowArray(TRKVOL)
        ShowArray(AVGSPD)
    end

//Read in variables from the network

    LENGTH = dview.(thefields[2])
    FC = dview.(thefields[3])
    M6FT = dview.(thefields[25])
    CO = s2i(dview.(thefields[26]))
    if recno = 1 then do
        ShowMessage("LENGTH = " + r2s(LENGTH))
        ShowMessage("FC = " + i2s(FC))
        ShowMessage("M6FT = " + i2s(M6FT))
        ShowMessage("CO = " + i2s(CO))
    end

//Skip Centroid Connectors
    if FC = 99 then goto Skip

//Do only Vandeburgh Co.
    if CO <> 82 then goto Skip

//Calculate hourly VMT by mode

```

```

        for i = 1 to 24 do
            C_VMT[i] = LENGTH * CARVOL[i]
            T_VMT[i] = LENGTH * TRKVOL[i]
        end
        VMT = {C_VMT, T_VMT}

//Calculate hourly VMT by mode by M6 facility type

        for m = 1 to 2 do
            for h = 1 to 24 do
                for f = 1 to 4 do
                    if M6FT = f then do
                        FVMT1[m][h][f] = FVMT1[m][h][f] + VMT[m][h]
                        FVMT1[m][h][5] = FVMT1[m][h][5] + VMT[m][h]
                    end
                end
            end
        end

//Tabulate Total VMT by hour

        for i = 1 to 24 do
            HVMT1[1][i] = HVMT1[1][i] + C_VMT[i] + T_VMT[i]
            if M6FT = 1 then do
                HVMT1[2][i] = HVMT1[2][i] + C_VMT[i] + T_VMT[i]
            end
            if M6FT = 2 then do
                HVMT1[3][i] = HVMT1[3][i] + C_VMT[i] + T_VMT[i]
            end
        end

//Calculate VMT by M6 Speed Bin by hour, by facility type

        for f = 1 to 2 do
            if M6FT = f then do
                for h = 1 to 24 do
                    z = AVGSPD[h]
                    if M6FT = 1 then z = (0.92*AVGSPD[h])-0.00213
                    s = 0
                    u = 0
                    while u = 0 do
                        s = s + 1
                        if z < M6spdbin[s] then u = s
                    end
                    l = u - 1
                    if l = 0 then l = 1
                    su = M6spdbin[u]
                    sl = M6spdbin[l]
                    if sl <> su then x = ((1/z) - (1/su))/((1/sl) - (1/su))
                    if sl = su then x = 1
                    SVM1[f][h][l] = SVM1[f][h][l] + (x*(C_VMT[h] + T_VMT[h]))
                    SVM1[f][h][u] = SVM1[f][h][u] + ((1-x)*(C_VMT[h] + T_VMT[h]))
                    SVM1[f][h][15] = SVM1[f][h][15] + C_VMT[h] + T_VMT[h]
                end
            end
        end

Skip:
    rec = GetNextRecord(dview + "|", null, null)
end
//end loop through joined network

// ShowArray(FVMT1)
// ShowArray(HVMT1)
// ShowArray(SVMT1)

UpdateProgressBar("Writing Data to Files", 99)

//Manipulating FVMT to convert from model network shares to real network shares

FVMT2 = CopyArray(FVMT1)
for m = 1 to 2 do
    for h = 1 to 24 do
        FVMT2[m][h][1] = 0.92 * FVMT1[m][h][1]
        FVMT2[m][h][3] = 2.2225 * FVMT1[m][h][3]
        FVMT2[m][h][4] = 0.08 * FVMT1[m][h][1]
    end
end

//Breaking FVMT out into 28 vehicle classes

y = yn
for c = 1 to 28 do
    m = MODE[c]
    for h = 1 to 24 do
        for f = 1 to 5 do
            FVMT3[c][h][f] = FVMT2[m][h][f] * CLASSES[y][c]
        end
    end
end
// ShowArray(FVMT3)

//Converting FVMT from sums to normalized percents

FVMT4 = CopyArray(FVMT3)
for c = 1 to 28 do
    for h = 1 to 24 do
        total = FVMT4[c][h][5]
        counter = 0
        while total < 0.9995 | total > 1.0004 do

```

```

        if total = 0 then do
            FVMT4[c][h] = {0.342, 0.498, 0.130, 0.030, 1.000}
            total = 1.000
        end
        FVMT4[c][h][5] = 0
        for f = 1 to 4 do
            FVMT4[c][h][f] = Round(FVMT4[c][h][f]/total, 3)
            FVMT4[c][h][5] = FVMT4[c][h][5] + FVMT4[c][h][f]
        end
        ltotal = total
        total = FVMT4[c][h][5]
        if total = ltotal | counter = 98 then do
            FVMT4[c][h][5] = 0
            maxval = ArrayMax(FVMT4[c][h])
            p = ArrayPosition(FVMT4[c][h], {maxval},)
            FVMT4[c][h][p] = FVMT4[c][h][p] + 1 - total
            total = 0
            for i = 1 to 4 do
                total = total + FVMT4[c][h][i]
            end
        end
        counter = counter + 1
        if counter = 100 then do
            errormsg = "Error: Normalization failed at FVMT[" +
                i2s(c) + "][" + i2s(h) + "]!"
            ShowMessage(errormsg)
            ShowArray(FVMT4[c][h])
            reterr = reterr + 1
            goto quit
        end
    end
end
FVMT = CopyArray(FVMT4)
for c = 1 to 28 do
    for i = 1 to 6 do
        j = i + 18
        FVMT[c][j] = FVMT4[c][i]
    end
    for i = 7 to 24 do
        j = i - 6
        FVMT[c][j] = FVMT4[c][i]
    end
end
ShowMessage("FVMT Finished!")
ShowArray(FVMT)

//Manipulating HVMT to reflect M6 numbering of the hours

TOTVMT = {0, 0, 0}
for q = 1 to 3 do
    for i = 1 to 6 do
        j = i + 18
        HVMT2[q][j] = HVMT1[q][i]
        TOTVMT[q] = TOTVMT[q] + HVMT1[q][i]
    end
    for i = 7 to 24 do
        j = i - 6
        HVMT2[q][j] = HVMT1[q][i]
        TOTVMT[q] = TOTVMT[q] + HVMT1[q][i]
    end
end
ShowArray(HVMT2)

//Calculating totals by speed bin by facility type and reformatting to create HBVMT

dim STOTALS[2]
STOTALS[1] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
STOTALS[2] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
for f = 1 to 2 do
    for s = 1 to 14 do
        for h = 1 to 24 do
            STOTALS[f][s] = STOTALS[f][s] + SVMT1[f][h][s]
            HBVMT[f][s][h] = SVMT1[f][h][s]
        end
    end
end

//Converting HVMT from sums to normalized percents

HVMT = CopyArray(HVMT2)
for q = 1 to 3 do
    total = TOTVMT[q]
    counter = 0
    while total < 0.99995 | total > 1.00004 do
        TOTVMT[q] = 0
        for i = 1 to 24 do
            HVMT[q][i] = Round(HVMT[q][i]/total, 4)
            TOTVMT[q] = TOTVMT[q] + HVMT[q][i]
        end
        ltotal = total
        total = TOTVMT[q]
        if total = ltotal | counter = 98 then do
            maxval = ArrayMax(HVMT[q])
            p = ArrayPosition(HVMT[q], {maxval},)
            HVMT[q][p] = HVMT[q][p] + 1 - total
            total = 0
            for i = 1 to 24 do
                total = total + HVMT[q][i]
            end
        end
    end

```

```

        counter = counter + 1
        if counter = 100 then do
            errmsg = "Error: Normalization failed on HVMT!"
            ShowMessage(errmsg)
            ShowArray(HVMT)
            reterr = reterr + 1
            goto quit
        end
    end
end
ShowMessage("HVMT Finished!")
ShowArray(HVMT)

//Converting SVMT from sums to normalized percents

SVMT3 = CopyArray(SVMT1)
for i = 1 to 2 do
    for j = 1 to 24 do
        total = SVMT3[i][j][15]
        counter = 0
        while total < 0.99995 | total > 1.00004 do
            SVMT3[i][j][15] = 0
            for k = 1 to 14 do
                SVMT3[i][j][k] = Round(SVMT3[i][j][k]/total, 4)
                SVMT3[i][j][15] = SVMT3[i][j][15] + SVMT3[i][j][k]
            end
            total = total
            total = SVMT3[i][j][15]
            if total = ltotal | counter = 97 then do
                SVMT3[i][j][15] = 0
                maxval = ArrayMax(SVMT3[i][j])
                p = ArrayPosition(SVMT3[i][j], {maxval},)
                SVMT3[i][j][p] = SVMT3[i][j][p] + 1 - total
                total = 0
                for k = 1 to 14 do
                    total = total + SVMT3[i][j][k]
                end
            end
            counter = counter + 1
            if counter = 100 then do
                errmsg = "Error: Normalization failed at SVMT[" +
                        i2s(i) + "][" + i2s(j) + "]!"
                ShowMessage(errmsg)
                ShowArray(SVMT[i][j])
                reterr = reterr + 1
                goto quit
            end
        end
    end
end
SVMT = CopyArray(SVMT3)
for f = 1 to 2 do
    for i = 1 to 6 do
        j = i + 18
        SVMT[f][j] = SVMT3[f][i]
    end
    for i = 7 to 24 do
        j = i - 6
        SVMT[f][j] = SVMT3[f][i]
    end
end
ShowMessage("SVMT Finished!")
ShowArray(SVMT)

//Creating HFTSVMT by normalizing SVMT1 by hour rather than by speed bin

HFTSVMT1 = CopyArray(HBVMT)
for f = 1 to 2 do
    for s = 1 to 14 do
        total = STOTALS[f][s]
        counter = 0
        while total < 0.999995 | total > 1.000004 do
            if total = 0 then do
                HFTSVMT1[f][s] = HVMT[1]
                total = 1.0000
                goto nexts
            end
            STOTALS[f][s] = 0
            for i = 1 to 24 do
                HFTSVMT1[f][s][i] = Round(HFTSVMT1[f][s][i]/total, 5)
                if HFTSVMT1[f][s][i] = 0 then HFTSVMT1[f][s][i] = 0.00001
                STOTALS[f][s] = STOTALS[f][s] + HFTSVMT1[f][s][i]
            end
            total = Round(total, 5)
            ltotal = total
            total = STOTALS[f][s]
            if total = ltotal | counter = 95 then do
                maxval = ArrayMax(HFTSVMT1[f][s])
                p = ArrayPosition(HFTSVMT1[f][s], {maxval},)
                HFTSVMT1[f][s][p] = HFTSVMT1[f][s][p] + 1 - total
                total = 0
                for i = 1 to 24 do
                    total = total + HFTSVMT1[f][s][i]
                end
            end
            counter = counter + 1
            if counter = 100 then do
                errmsg1 = "Error: Normalization failed on HBVMT at s = "
                errmsg2 = i2s(s) + "!"
                ShowMessage(errmsg1 + errmsg2)
                ShowArray(HBVMT)
            end
        end
    end
end

```

```

        ShowArray(HFTSVMT1)
        ShowArray(STOTALS)
        reterr = reterr + 1
        goto quit
    end
    nexts:
    end
end
HFTSVMT = CopyArray(HFTSVMT1)
for f = 1 to 2 do
for s = 1 to 14 do
    for i = 1 to 6 do
        j = i + 18
        HFTSVMT[f][s][j] = HFTSVMT1[f][s][i]
    end
    for i = 7 to 24 do
        j = i - 6
        HFTSVMT[f][s][j] = HFTSVMT1[f][s][i]
    end
end
end

//Write FVMT M6 input file

ffile = OpenFile(theopath + "\\post\\FVMT.d", "w")
WriteLine(ffile, "VMT BY FACILITY")
WriteLine(ffile, "* VMT fractions are listed for 28 vehicle classes.")
for i = 1 to 28 do
    fsthr = " "
    if i < 10 then fsthr = fsthr + " "
    fsthr = fsthr + i2s(i)
    for k = 1 to 4 do
        plusnext = r2s(FVMT[i][1][k])
        while Len(plusnext)<5 do
            plusnext = plusnext + "0"
        end
        fsthr = fsthr + " " + plusnext
    end
    WriteLine(ffile, fsthr)
    for j = 2 to 24 do
        hrvmt = " "
        for k = 1 to 4 do
            plusnext = r2s(FVMT[i][j][k])
            while Len(plusnext)<5 do
                plusnext = plusnext + "0"
            end
            hrvmt = hrvmt + plusnext + " "
        end
        WriteLine(ffile, hrvmt)
    end
end
CloseFile(ffile)

//Write FVMT M6 input file for each facility type

for f = 1 to 4 do
ffile = OpenFile(theopath + "\\post\\FVMT" + i2s(f) + ".d", "w")
WriteLine(ffile, "VMT BY FACILITY")
WriteLine(ffile, "* VMT fractions are listed for 28 vehicle classes.")
for i = 1 to 28 do
    fsthr = " "
    if i < 10 then fsthr = fsthr + " "
    fsthr = fsthr + i2s(i)
    for k = 1 to 4 do
        plusnext = "0.000"
        if k = f then plusnext = "1.000"
        fsthr = fsthr + " " + plusnext
    end
    WriteLine(ffile, fsthr)
    for j = 2 to 24 do
        hrvmt = " "
        for k = 1 to 4 do
            plusnext = "0.000"
            if k = f then plusnext = "1.000"
            hrvmt = hrvmt + plusnext + " "
        end
        WriteLine(ffile, hrvmt)
    end
end
CloseFile(ffile)
end

//Write HVMT M6 input file

for q = 1 to 3 do
hfile = OpenFile(theopath + "\\post\\HVMT" + i2s(q) + ".d", "w")
WriteLine(hfile, "VMT BY HOUR")
WriteLine(hfile, "* Fraction of all vehicle miles traveled by hour of the day.")
WriteLine(hfile, "* First hour is 6 a.m.")
WriteLine(hfile, "* ")
for i = 1 to 4 do
    line = " "
    for j = 1 to 6 do
        h = (i-1)*6 + j
        plusnext = r2s(HVMT[q][h])
        while Len(plusnext)<6 do
            plusnext = plusnext + "0"
        end
        line = line + plusnext + " "
    end

```

```

        end
        WriteLine(hfile, line)
    end
    CloseFile(hfile)
end

//Write HVMT M6 input file for each speed bin and facility type

for f = 1 to 2 do
for s = 1 to 14 do
hfile = OpenFile(thepath + "\\post\\HVMT" + i2s(f) + "_" + i2s(s) + ".d", "w")
WriteLine(hfile, "VMT BY HOUR")
WriteLine(hfile, "* Fraction of all vehicle miles traveled by hour of the day.")
WriteLine(hfile, "* First hour is 6 a.m.")
WriteLine(hfile, "* ")
for i = 1 to 4 do
    line = "      "
    for j = 1 to 6 do
        h = (i-1)*6 + j
        plusnext = r2s(HFTSVMT[f][s][h])
        if plusnext = "1" then plusnext = "1.0000"
        while Len(plusnext)<6 do
            plusnext = plusnext + "0"
        end
        line = line + plusnext + "  "
    end
    WriteLine(hfile, line)
end
CloseFile(hfile)
end
end

//Write SVMT M6 input file for each speed bin

for s = 1 to 14 do
sfile = OpenFile(thepath + "\\post\\SVMT" + i2s(s) + ".d", "w")
WriteLine(sfile, "SPEED VMT")
for i = 1 to 2 do
    for j = 1 to 24 do
        line = i2s(i) + " " + i2s(j) + " "
        if j < 10 then line = line + " "
        for k = 1 to 14 do
            plusnext = "0.0000"
            if k = s then plusnext = "1.0000"
            line = line + plusnext + " "
        end
        WriteLine(sfile, line)
    end
    CloseFile(sfile)
end

//Write SVMT M6 input file

sfile = OpenFile(thepath + "\\post\\SVMT.d", "w")
WriteLine(sfile, "SPEED VMT")
for i = 1 to 2 do
    for j = 1 to 24 do
        line = i2s(i) + " " + i2s(j) + " "
        if j < 10 then line = line + " "
        for k = 1 to 14 do
            plusnext = r2s(SVMT[i][j][k])
            while Len(plusnext)<6 do
                plusnext = plusnext + "0"
            end
            line = line + plusnext + " "
        end
        WriteLine(sfile, line)
    end
    CloseFile(sfile)

//Close the macro
quit:
    CloseView(dview)
    CloseView(mvw)

    DestroyProgressBar()
    SetStatus(1, "@System0", )
quit2:
    Return(reterr)
endMacro

Macro "M6in" (in_value)
    thepath = in_value[1]
    anyr = in_value[2]

/*
A Program written by Vincent L. Bernardin, Jr. of Bernardin, Lochmueller, and Assoc. to generate Mobile6 control (*.in) files for
Vanderburgh Co.
6/13/03
Revised 6/23/03
*/
gp = GetProgram()
TCpath = SplitPath(gp[1])
infile = OpenFile(TCpath[1] + TCpath[2] + "\\Mobile6.in", "w")
WriteLine(infile, "MOBILE6 INPUT FILE :")
WriteLine(infile, "*created by M6in macro written 6/13/03, vlb2")
WriteLine(infile, "")
WriteLine(infile, "POLLUTANTS : HC CO NOX")

```

```

WriteLine(infile, "REPORT FILE      : " + thepath + "\\post\\M6REPORT.txt")
WriteLine(infile, "SPREADSHEET    : M6REPORT")
WriteLine(infile, "")
WriteLine(infile, "RUN DATA")
WriteLine(infile, "MIN/MAX TEMP   : 68. 89.")
WriteLine(infile, "FUEL RVP       : 9.0")
WriteLine(infile, "")
WriteLine(infile, "REG DIST        : " + thepath + "\\post\\VREGDATA.D")
WriteLine(infile, "VMT FRACTIONS  :")
if anyr = 1 then yrvmf1 = "0.617  0.044  0.148  0.059  0.027  0.032  0.003  0.002"
if anyr = 1 then yrvmf2 = "0.002  0.007  0.008  0.009  0.033  0.002  0.001  0.006"
if anyr = 2 then yrvmf1 = "0.600  0.046  0.154  0.060  0.027  0.034  0.003  0.003"
if anyr = 2 then yrvmf2 = "0.002  0.008  0.009  0.010  0.035  0.002  0.001  0.006"
if anyr = 3 then yrvmf1 = "0.583  0.048  0.159  0.061  0.028  0.037  0.004  0.003"
if anyr = 3 then yrvmf2 = "0.002  0.008  0.010  0.011  0.038  0.002  0.001  0.005"
if anyr > 3 then yrvmf1 = "0.577  0.049  0.163  0.061  0.028  0.038  0.004  0.003"
if anyr > 3 then yrvmf2 = "0.002  0.008  0.010  0.011  0.039  0.002  0.001  0.004"
WriteLine(infile, yrvmf1)
WriteLine(infile, yrvmf2)
WriteLine(infile, "")
YEARS = {2000, 2006, 2015, 2025, 2030}
year = YEARS[anyr]
for f = 1 to 2 do
for s = 1 to 14 do
if f = 1 then ftype = "Freeways"
if f = 2 then ftype = "Arterials"
M6spdbins = {0, 2.5, 7.5, 12.5, 17.5, 22.5, 27.5, 32.5,
            37.5, 42.5, 47.5, 52.5, 57.5, 62.5, 500}
lows = r2s(M6spdbins[s])
highs = r2s(M6spdbins[s+1])
scnname = i2s(year) + " " + ftype + " " + lows + "mph to " + highs + "mph"
WriteLine(infile, "SCENARIO RECORD  : Scenario Title : Vanderburg Co., " + scnname)
WriteLine(infile, "CALENDAR YEAR   : " + i2s(year))
WriteLine(infile, "EVALUATION MONTH : 7")
WriteLine(infile, "CLOUD COVER    : 0.27")
WriteLine(infile, "SUNRISE/SUNSET  : 7 9")
WriteLine(infile, "VMT BY FACILITY : " + thepath + "\\post\\Fvmt" + i2s(f) + ".d")
WriteLine(infile, "VMT BY HOUR     : " + thepath + "\\post\\Hvmt" + i2s(f) + "_" + i2s(s) + ".d")
WriteLine(infile, "SPEED VMT      : " + thepath + "\\post\\Svmt" + i2s(s) + ".d")
WriteLine(infile, "")
end
end
for f = 3 to 4 do
if f = 3 then ftype = "Locals"
if f = 4 then ftype = "Ramps"
scnname = i2s(year) + " " + ftype
WriteLine(infile, "SCENARIO RECORD  : Scenario Title : Vanderburg Co., " + scnname)
WriteLine(infile, "CALENDAR YEAR   : " + i2s(year))
WriteLine(infile, "EVALUATION MONTH : 7")
WriteLine(infile, "CLOUD COVER    : 0.27")
WriteLine(infile, "SUNRISE/SUNSET  : 7 9")
WriteLine(infile, "VMT BY FACILITY : " + thepath + "\\post\\Fvmt" + i2s(f) + ".d")
WriteLine(infile, "VMT BY HOUR     : " + thepath + "\\post\\Hvmt1.d")
WriteLine(infile, "")
end
WriteLine(infile, "END OF RUN")
CloseFile(infile)

endMacro

Macro "RunM6" (in_value)
  thepath = in_value[1]

  ShowMessage("When Mobile6 prompts you for an input file, simply press ENTER.")

  SetEnvironmentVariable("PATH", thepath + "\\post")

  status = RunProgram("Mobile62.exe", )

endMacro

Macro "Apply" (in_value)
  thepath = in_value[1]
  linevw = in_value[2]
  anyr = in_value[3]
  aqscn = in_value[4]

  repfilename = thepath + "\\post\\M6report.txt"
  dbfname = thepath + "\\post\\M6rates.dbf"

  repfile = OpenFile(repfilename, "r")

  REPARR = ReadArray(repfile)

  dim VOC[30]
  dim CO[30]
  dim NOX[30]
  c = 0
  for i = 1 to REPARR.length do
    comp = Word(REPARR[i], 1)
    if comp = "Composite" then do
      emtype = Word(REPARR[i], 2)
      if emtype = "VOC" then do
        c = c + 1
        rate = Word(REPARR[i], 21) + "." + Word(REPARR[i], 22)
        VOC[c] = Round(Value(rate), 3)
      end
      if emtype = "CO" then do
        rate = Word(REPARR[i], 21) + "." + Word(REPARR[i], 22)
        CO[c] = Round(Value(rate), 3)
      end
    end
  end

```



```

VMT2 = 0
VMT3 = 0
VOC1 = 0
CO1 = 0
NOX1 = 0
VOC2 = 0
CO2 = 0
NOX2 = 0
dim CARVOL[24]
dim TRKVOL[24]
dim AVGSPD[24]
dim C_VMT[24]
dim T_VMT[24]

//Create the 24M6field arrays
dim carvolfield[24]
dim trkvolfield[24]
dim avgspdfield[24]
for i = 1 to 24 do
    if i > 9 then do
        carvolfield[i] = "CARVOL_" + i2s(i)
        trkvolfield[i] = "TRKVOL_" + i2s(i)
        avgspdfield[i] = "AVGSPD_" + i2s(i)
    end
    else do
        carvolfield[i] = "CARVOL_0" + i2s(i)
        trkvolfield[i] = "TRKVOL_0" + i2s(i)
        avgspdfield[i] = "AVGSPD_0" + i2s(i)
    end
end

//Create, Open, and begin Looping the Joined View

totrec = GetRecordCount(linevw, )
dview = JoinViews("Network+24M6", linevw + ".ID", mvw + ".LID", )

SetView(dview)
recno = 0
rec = GetFirstRecord(dview + "|", null)
while rec <> null do

    recno = recno + 1
    prog = Round ((recno/totrec)*99, 0)
    UpdateProgressBar("Computing Emissions", prog)
    SetStatus(1, "Record " + i2s(recno) + " of " + i2s(totrec), )

//Read in 24M6 data arrays

    for i = 1 to 24 do
        CARVOL[i] = dview.(carvolfield[i])
        TRKVOL[i] = dview.(trkvolfield[i])
        AVGSPD[i] = dview.(avgspdfield[i])
    end

//Read in variables from the network

    LENGTH = dview.(thefields[2])
    FC = dview.(thefields[3])
    DTV = dview.(thefields[17])
    M6FT = dview.(thefields[25])
    CCounty = s2i(dview.(thefields[26]))

//Skip Centroid Connectors
    if FC = 99 then goto Skip

//Do only Vandeburgh Co.
    if CCounty <> 82 then goto Skip

//Calculate hourly VMT by mode

    for i = 1 to 24 do
        C_VMT[i] = LENGTH * CARVOL[i]
        T_VMT[i] = LENGTH * TRKVOL[i]
        VMT[i] = C_VMT[i] + T_VMT[i]
    end

//Sum Local VMT

    if M6FT = 3 | M6FT = 4 then do
        for i = 1 to 24 do
            VMT3 = VMT3 + VMT[i]
        end
        goto Skip
    end

//Initialize Variables

    VOC24hr = 0
    CO24hr = 0
    NOX24hr = 0
    VMT24hr = 0

//Apply Rates to Freeways

    if M6FT = 1 then do
        for i = 1 to 24 do
            z = (0.92*AVGSPD[i])-0.00213
            s = 0
            u = 0
            while u = 0 do
                s = s + 1

```

```

        if z < M6spdbin[s] then u = s
    end
    l = u - 1
    if l = 0 then l = 1
    su = M6spdbin[u]
    sl = M6spdbin[l]
    if sl <> su then x = ((1/z) - (1/su))/((1/sl) - (1/su))
    if sl = su then x = 1
    hrVOC[i] = x*VOC[1]*VMT[i] + (1-x)*VOC[u]*VMT[i]
    hrCO[i] = x*CO[1]*VMT[i] + (1-x)*CO[u]*VMT[i]
    hrNOX[i] = x*NOX[1]*VMT[i] + (1-x)*NOX[u]*VMT[i]
    VOC24hr = VOC24hr + hrVOC[i]
    CO24hr = CO24hr + hrCO[i]
    NOX24hr = NOX24hr + hrNOX[i]
    VMT24hr = VMT24hr + VMT[i]
end

dview.VOC = VOC24hr
dview.CO1 = CO24hr
dview.NOX = NOX24hr
VOC1 = VOC1 + (0.92 * VOC24hr)
CO1 = CO1 + (0.92 * CO24hr)
NOX1 = NOX1 + (0.92 * NOX24hr)
VMT1 = VMT1 + VMT24hr

//Apply Rates to Arterials/Collectors

if M6FT = 2 then do
    for i = 1 to 24 do
        z = AVGSPD[i]
        s = 0
        u = 0
        while u = 0 do
            s = s + 1
            if z < M6spdbin[s] then u = s
        end
        l = u - 1
        if l = 0 then l = 1
        su = M6spdbin[u]
        sl = M6spdbin[l]
        if sl <> su then x = ((1/z) - (1/su))/((1/sl) - (1/su))
        if sl = su then x = 1
        l = l + 14
        u = u + 14
        hrVOC[i] = x*VOC[1]*VMT[i] + (1-x)*VOC[u]*VMT[i]
        hrCO[i] = x*CO[1]*VMT[i] + (1-x)*CO[u]*VMT[i]
        hrNOX[i] = x*NOX[1]*VMT[i] + (1-x)*NOX[u]*VMT[i]
        VOC24hr = VOC24hr + hrVOC[i]
        CO24hr = CO24hr + hrCO[i]
        NOX24hr = NOX24hr + hrNOX[i]
        VMT24hr = VMT24hr + VMT[i]
    end
    dview.VOC = VOC24hr
    dview.CO1 = CO24hr
    dview.NOX = NOX24hr
    VOC2 = VOC2 + VOC24hr
    CO2 = CO2 + CO24hr
    NOX2 = NOX2 + NOX24hr
    VMT2 = VMT2 + VMT24hr
end

Skip:
rec = GetNextRecord(dview + "|", null, null)
end
//end loop through joined network
UpdateProgressBar("Writing Report File", 99)

//Apply Rate to Local VMT factored to HPMS

VMT3 = 2.2225 * VMT3
VOC3 = VOC[29] * VMT3
CO3 = CO[29] * VMT3
NOX3 = NOX[29] * VMT3

//Apply Rate for Ramps

VMT4 = 0.08 * VMT1
VMT1 = 0.92 * VMT1
VOC4 = VOC[30] * VMT4
CO4 = CO[30] * VMT4
NOX4 = NOX[30] * VMT4

VandVMT = VMT1 + VMT2 + VMT3 + VMT4

//Combine functional types to get totals and convert from kg to tons

VOCKg = (VOC1 + VOC2 + VOC3 + VOC4)/1000
COKg = (CO1 + CO2 + CO3 + CO4)/1000
NOXkg = (NOX1 + NOX2 + NOX3 + NOX4)/1000

VOCton = VOCKg / 907.813
Coton = COKg / 907.813
NOXton = NOXkg / 907.813

//Temporarily use Notes to report rather than file
//ShowMessage("VOC = " + r2s(VOCTon) + " tons")

```

```

// ShowMessage("CO = " + r2s(COton) + " tons")
// ShowMessage("NOX = " + r2s(NOXton) + " tons")

//Write Report File

opt = {{"Initial Directory", thepath + "\\post\\\"},  

       {"Suggested Name", "AQreport"},}  

outfile = ChooseFileName({{"text", "*.txt"}}, "Choose Report File", opt)

YEARS = {2000, 2006, 2015, 2025, 2030}
year = YEARS[anyr]

budget = "Budget:      10.91 tons/day    77.94 tons/day    11.56 tons/day"  

if anyr = 1 then budget = "Budget:      16.29 tons/day    106.96 tons/day    12.52 tons/day"

ofile = OpenFile(outfile, "w")
WriteLine(ofile, "Air Quality Conformity Analysis Report for Vanderburgh County")
WriteLine(ofile, "from MOBILE6 and the Evansville Regional Travel Model")
datetime = GetDateAndTime()
WriteLine(ofile, datetime)
WriteLine(ofile, " ")
WriteLine(ofile, " ")
WriteLine(ofile, "Year: " + i2s(year))
WriteLine(ofile, "Scenario: " + aqscn)
WriteLine(ofile, " " + r2s(Round(VandVMT, 0)) + " VMT in Vanderburgh County")
WriteLine(ofile, " ")
WriteLine(ofile, "           VOC          CO          NOX")
//This line is simply for visual      xx.xx tons/day   xxx.xx tons/day   xx.xx tons/day
VOCip = Round(VOCTon, 0)
COip = Round(COton, 0)
NOXip = Round(NOXton, 0)
VOCdp = r2s(Round(VOCTon - VOCip, 2))
CODp = r2s(Round(COton - COip, 2))
NOXdP = r2s(Round(NOXton - NOXip, 2))
VOCip = r2s(VOCip)
COip = r2s(COip)
NOXip = r2s(NOXip)
if Len(VOCip) < 2 then VOCip = " " + VOCip
while Len(COip) < 3 do
    COip = " " + COip
end
if Len(NOXip) < 2 then NOXip = " " + NOXip
while Len(VOCdp) < 4 do
    if Len(VOCdp) = 1 then VOCdp = VOCdp + "."
    VOCdp = VOCdp + "0"
end
while Len(CODp) < 4 do
    if Len(CODp) = 1 then CODp = CODp + "."
    CODp = CODp + "0"
end
while Len(NOXdp) < 4 do
    if Len(NOXdp) = 1 then NOXdP = NOXdP + "."
    NOXdP = NOXdP + "0"
end
VOCdp = Right(VOCdp, 3)
CODp = Right(CODp, 3)
NOXdP = Right(NOXdp, 3)
VOCstr = VOCip + VOCdp + " tons/day  "
COstr = COip + CODp + " tons/day  "
NOXstr = NOXip + NOXdP + " tons/day"
WriteLine(ofile, "Scenario: " + VOCstr + COstr + NOXstr)
WriteLine(ofile, budget)
CloseFile(ofile)

//Close the macro
quit:
    CloseView(dview)
    CloseView(mvw)

DestroyProgressBar()
SetStatus(1, "@System0", )
quit2:
endMacro

Macro "add_fields" (in_value)
dview = in_value[1]
fldnames = in_value[2]
typeflags = in_value[3]

/*
A utility macro written by Vince Bernardin, Jr. 5/15/03
updated 6/18/03
*/
fd = fldnames.length
dim fldtypes(fd)
for i = 1 to fldnames.length do
    if typeflags[i] = "r" then fldtypes[i] = {"Real", 10, 4}
    if typeflags[i] = "i" then fldtypes[i] = {"Integer", 10, 3}
    if typeflags[i] = "c" then fldtypes[i] = {"String", 16, null}
end

SetView(dview)

struct = GetTableStructure(dview)
dim snames[1]
for i = 1 to struct.length do

```

```

    struct[i] = struct[i] + {struct[i][1]}
    snames = snames + {struct[i][1]}
end

    modtab = 0
for i = 1 to fldnames.length do
    pos = ArrayPosition(snames, {fldnames[i]}, )
    if pos = 0 then do
        newstr = newstr + {{fldnames[i], fldtypes[i][1], fldtypes[i][2], fldtypes[i][3],
                           "false", null, null, null, null}}
    modtab = 1
    end
end

if modtab = 1 then do
    newstr = struct + newstr
    ModifyTable(dview, newstr)
end

endMacro

```